

Оглавление

История изменений.....	1
Настройки компьютера с СУБД (MySQL/PostgreSQL).....	2
Установка и настройка СУБД MySQL.....	3
Установка СУБД MySQL Версии 5.7.44	9
Создание БД для контроля КМ	11
Запуск утилиты MySQL Command Line Client.....	11
Создание БД для контроля КМ	13
Установка и настройка СУБД PostgreSQL	14
Установка СУБД PostgreSQL	14
Создание БД для контроля КМ	20
Запуск утилиты SQL Shell (psql).....	20
Создание БД для контроля КМ	23
Дополнительные настройки для удаленного доступа к SQL-серверу	26
Настройка правил брандмауэра	26
Настройка SQL-сервера для прослушивания внешних подключений	28
MySQL	28
PostgreSQL.....	30
Настройка прав пользователя для удаленного подключения	33
MySQL	33
PostgreSQL.....	36
Настройки компьютера с ПО “Кассир 5”	39
Установка ODBC-драйвера.....	39
Установка ODBC-драйвера MySQL	39
Установка ODBC-драйвера PostgreSQL (psqlodbc)	41
Настройка источников данных ODBC	42
Настройка источника данных MySQL	44
Настройка источника данных PostgreSQL.....	45
Настройка ПО Кассир.....	46

История изменений

- 08.04.2024 – Версия 2
Исправлен текст запроса создания таблиц в PostgreSQL.
- 14.05.2024 – Версия 3
Доработана инструкцию по настройке удаленного доступа к SQL-серверу

Настройки компьютера с СУБД (MySQL/PostgreSQL)

Обратите внимание!

Если SQL-сервер, который вы собираетесь использовать для контроля КМ по кранам **УЖЕ** используется для чего-то кроме нужд ПО Кассир, то рекомендуется обратиться к ответственному за администрирование данного сервера с запросом о выделении:

1. отдельной базы для контроля км по кранам.
2. отдельного пользователя СУБД (или нескольких пользователей - для каждой кассы отдельного) для доступа к этой базе. При этом хотя бы у одного пользователя должна быть возможность создавать таблицы в этой базе.
3. при этом необходимо сообщить количество касс, которые будут использовать контроль КМ по кранам, так как они могут одновременно подключиться к серверу, а количество подключений в один момент времени ограничено.

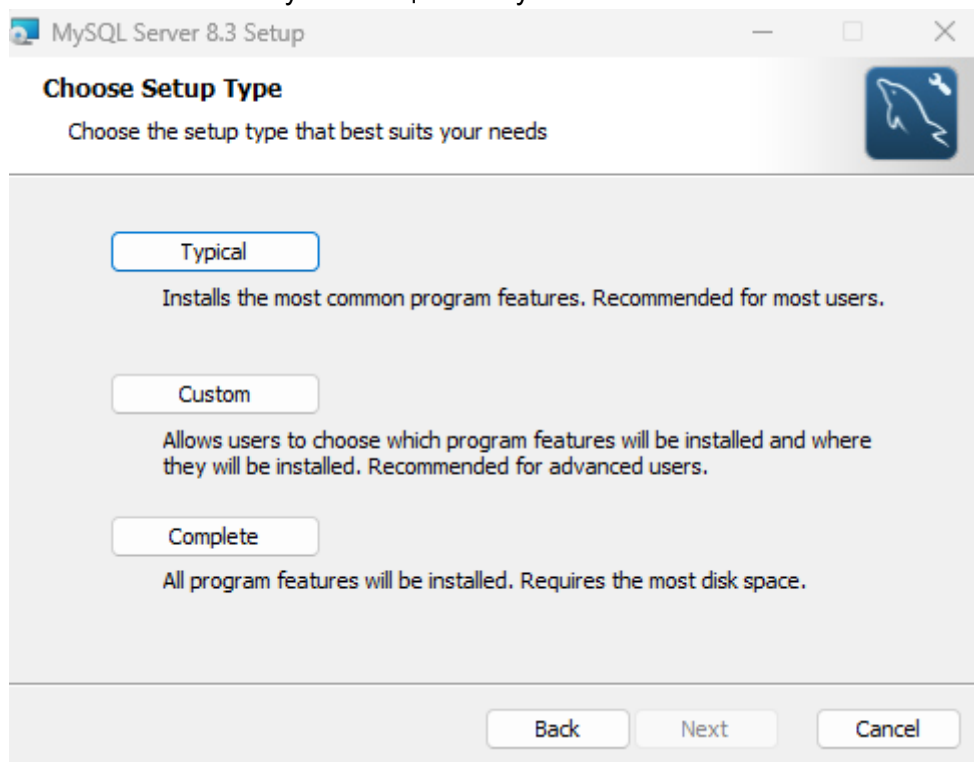
Установка и настройка СУБД MySQL

Если есть опыт в установке и настройке MySQL, то этот раздел будет мало полезен, так как в установке MySQL для работы контроля КМ нет ничего специфического.

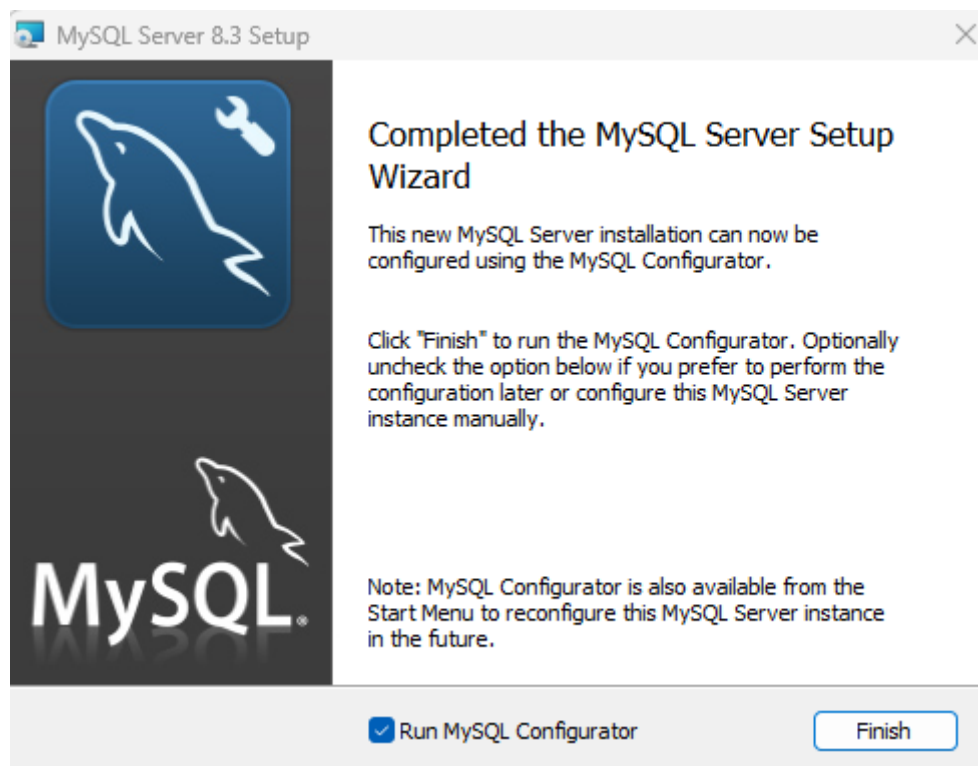
Для начала надо получить установщик MySQL. Для этого переходим по ссылке: <https://dev.mysql.com/downloads/mysql/> или <https://dev.mysql.com/downloads/windows/installer>.

Далее описание процесса установки MySQL будет сопровождаться скриншотами, актуальными для MySQL 8.3 на примере установки в Windows 11.

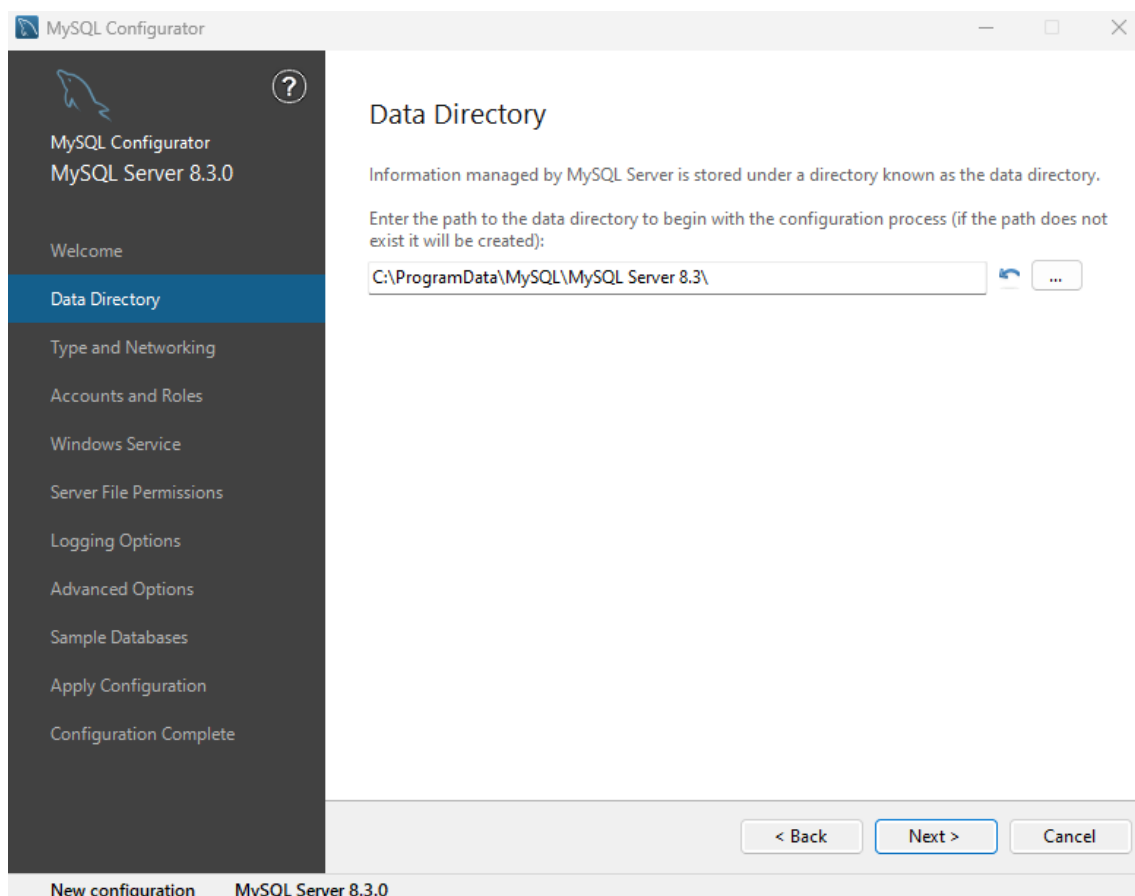
После скачивания установщика запускаем его.



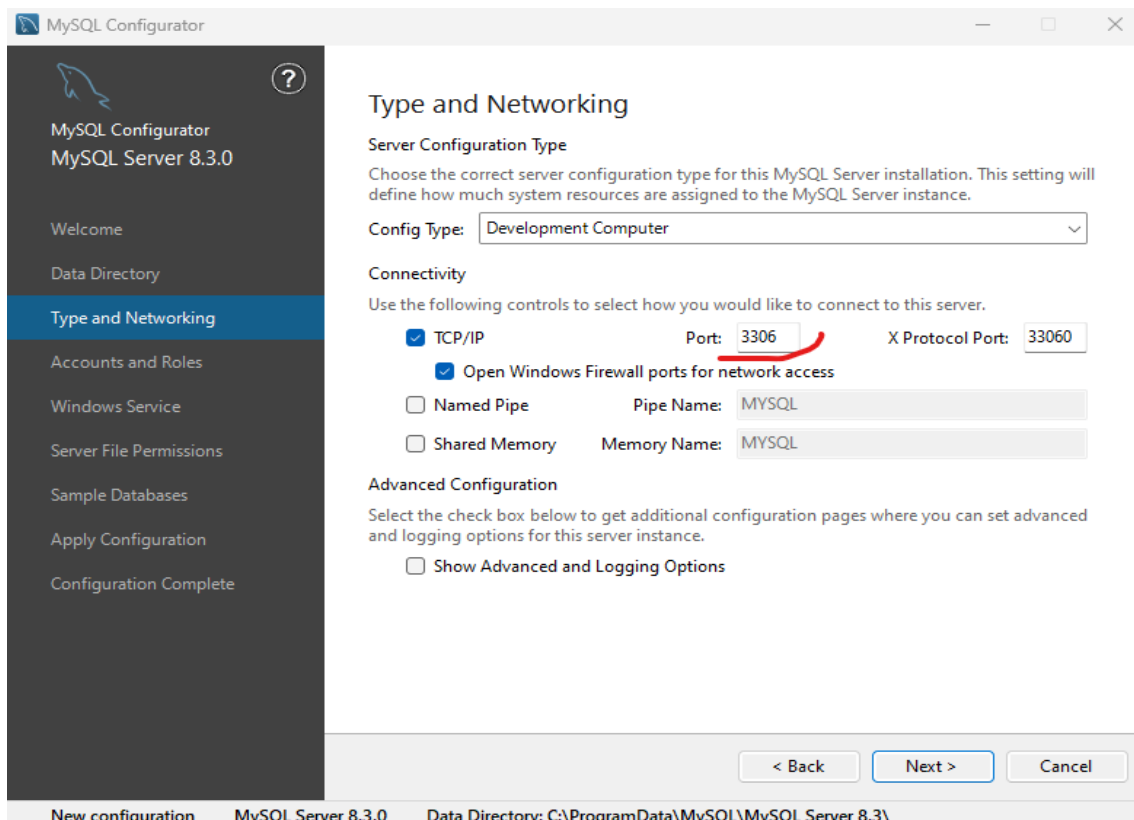
На этом этапе выбираем **Complete** и нажимаем **Next**.



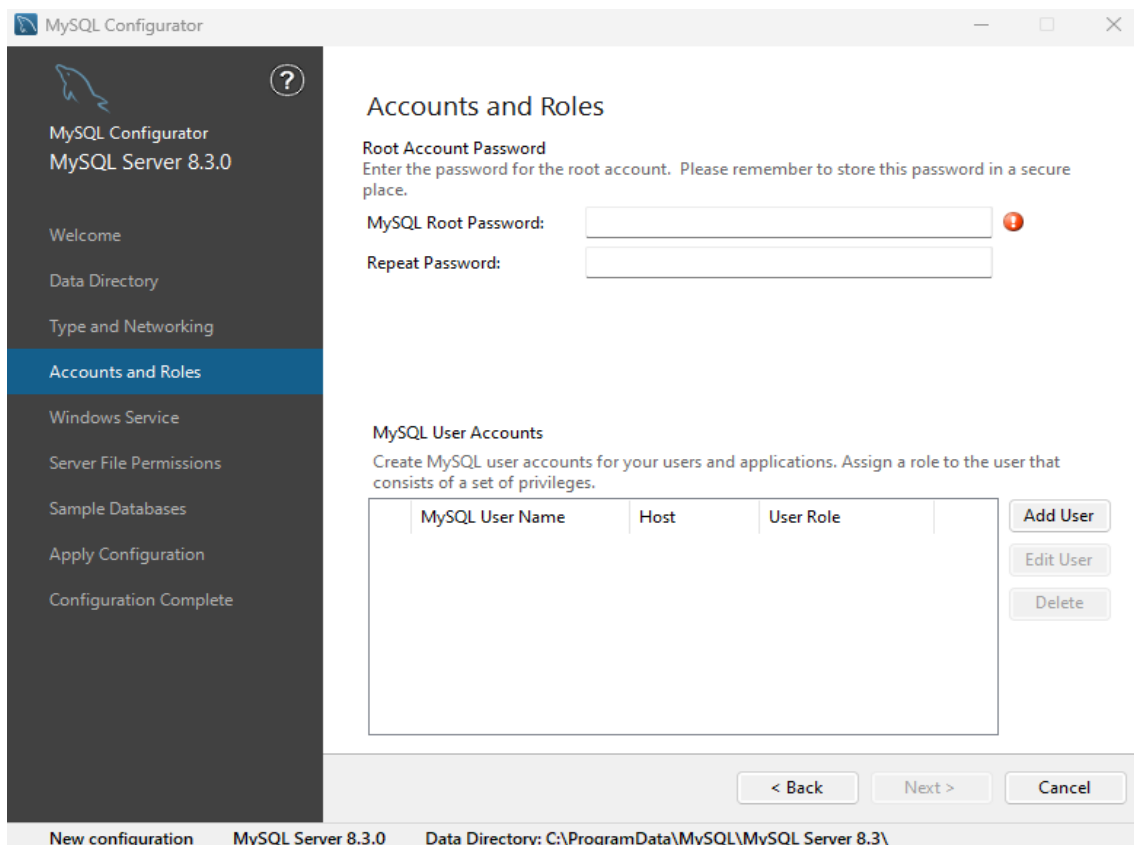
На этом этапе обязательно ставим галочку **Run MySQL Configuration** и жмем **Finish**.



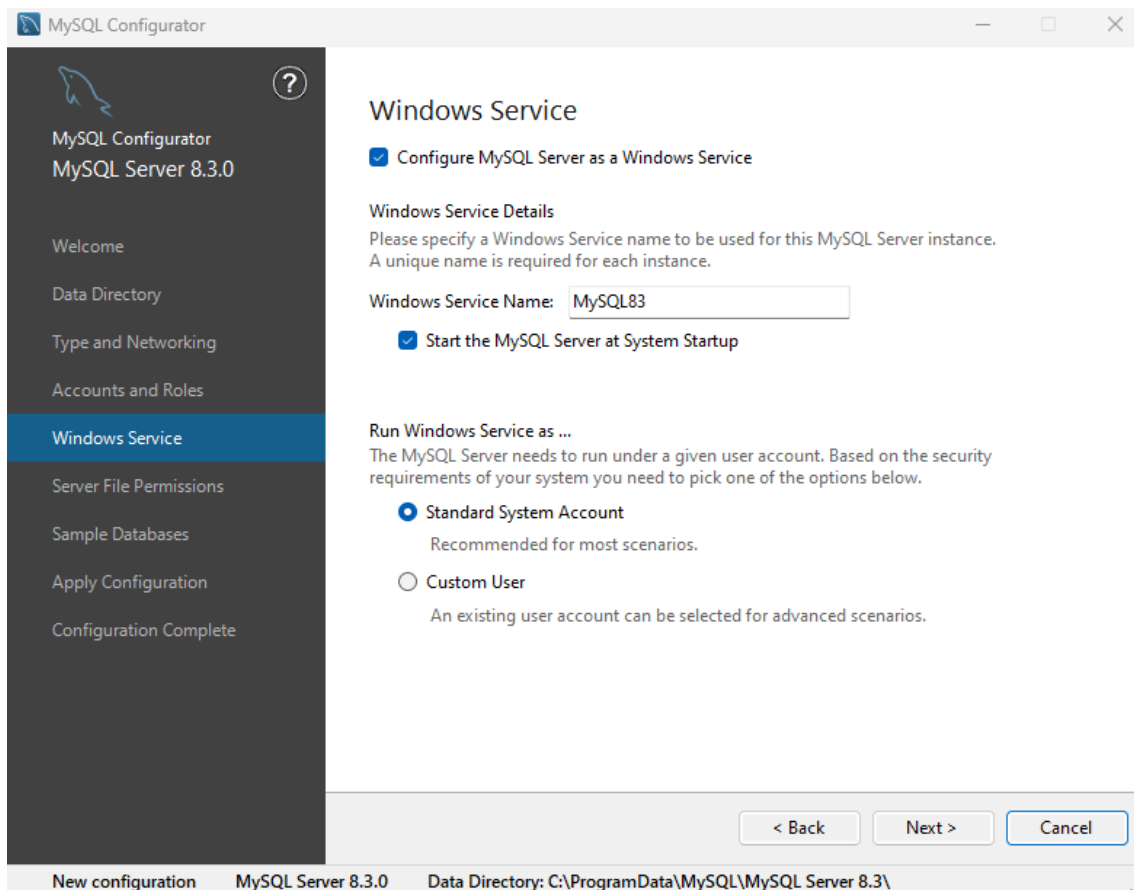
Запущена программа конфигурирования СУБД. На данном этапе указываем папку для установки СУБД и жмем **Next >**.



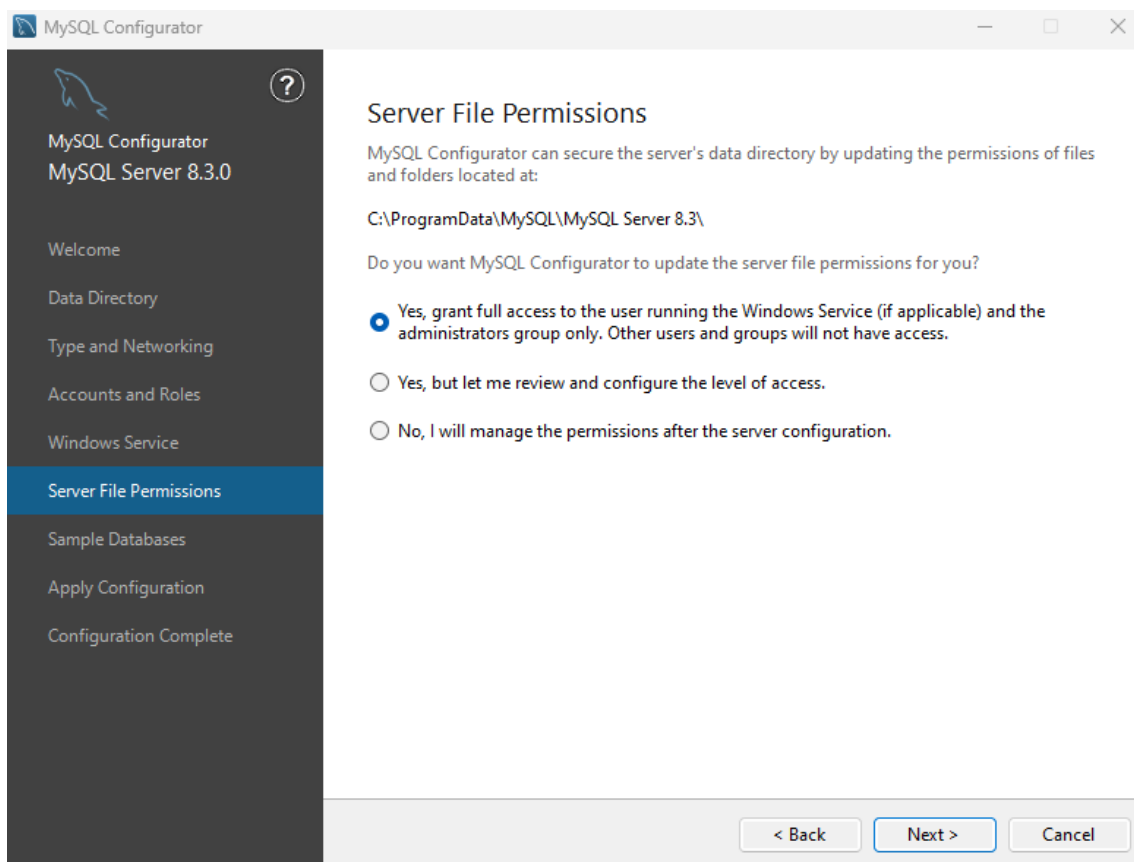
На данном этапе оставляем галочки как на скриншоте и указываем номер порта, который СУБД будет использовать для прослушивания запросов подключения.



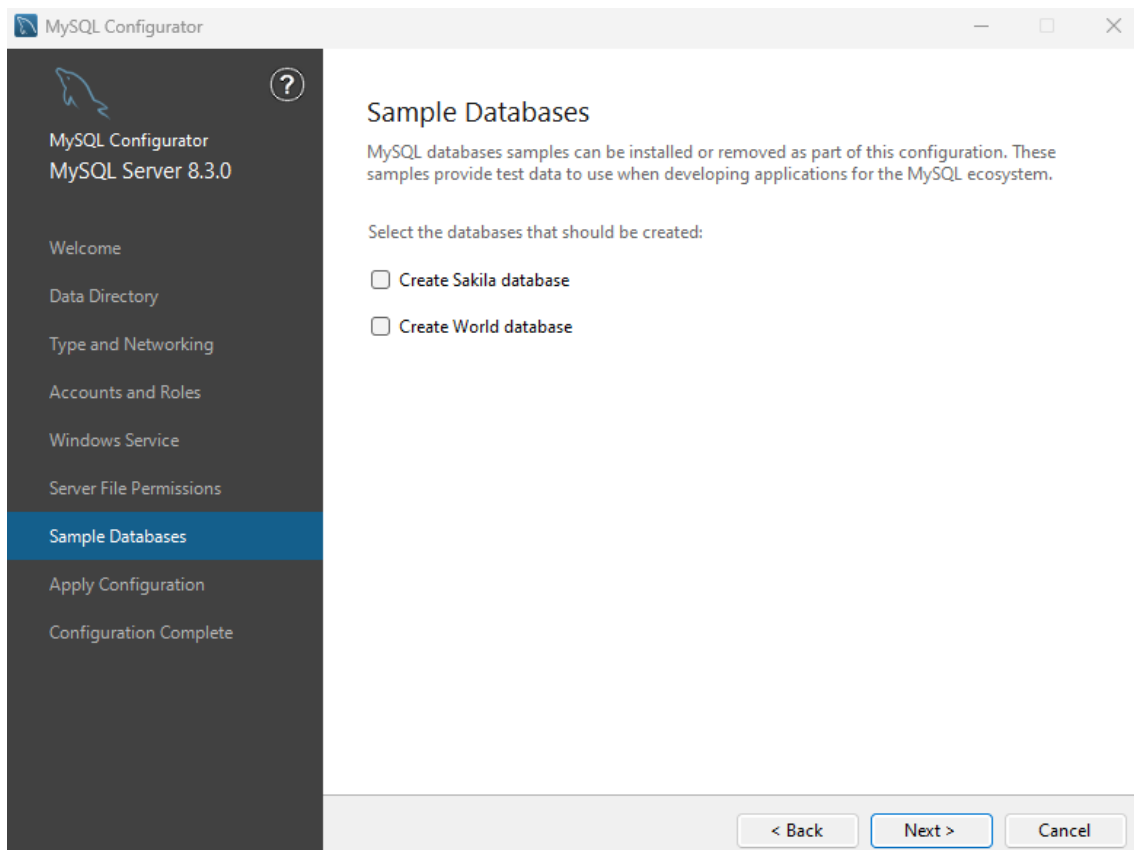
Тут указываем пароль супер пользователя и нажимаем **Next >**.



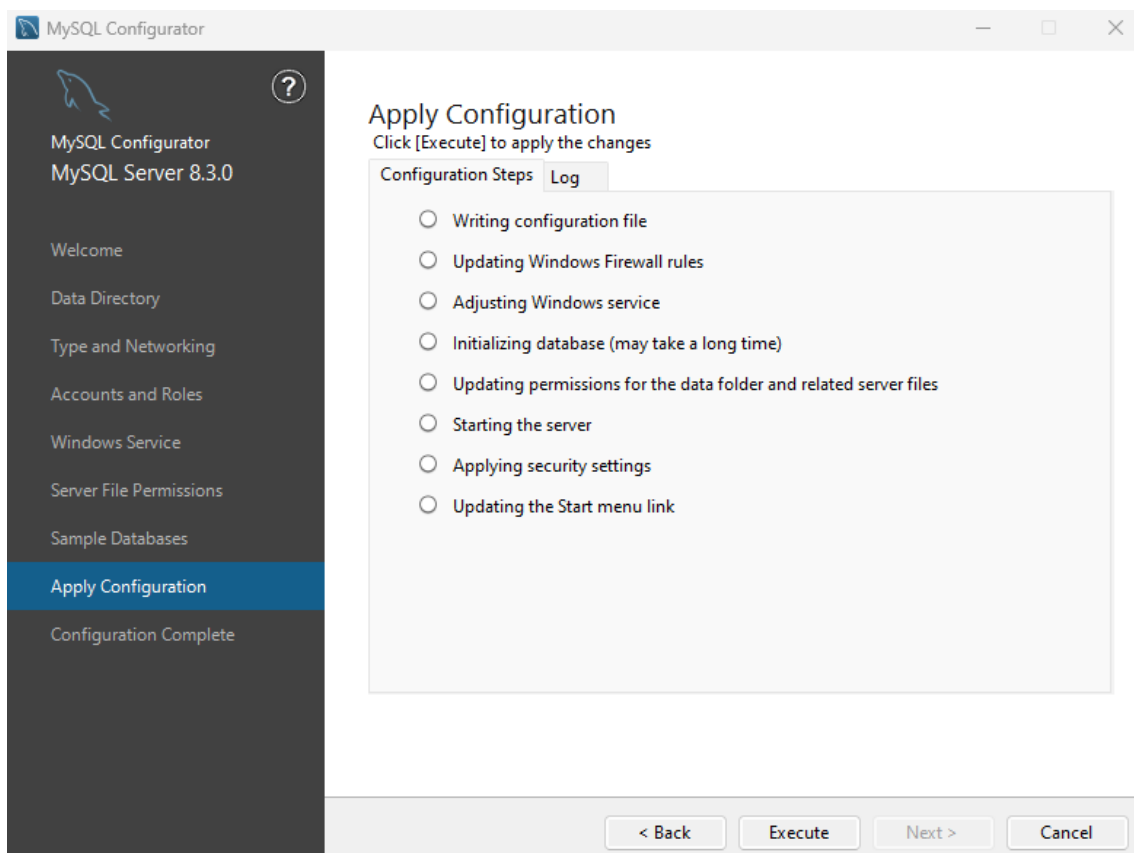
Тут оставляем как есть и жмем **Next >**.



Тут оставляем как есть и жмем **Next >**.



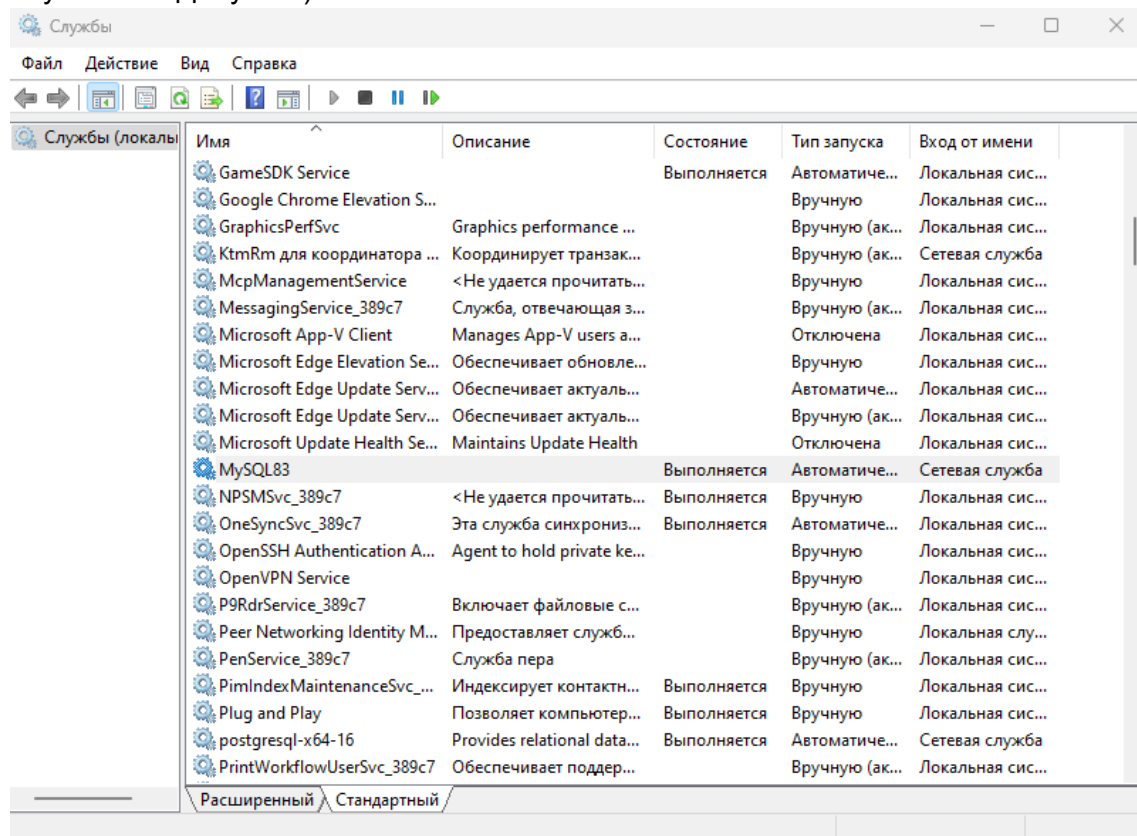
Тут оставляем как есть и жмем **Next >**.



Жмем **Execute**.

Для проверки правильности установки надо/можно проверить список служб Windows (если все сделано по инструкции, то должна была создаваться

служба СУБД MySQL):

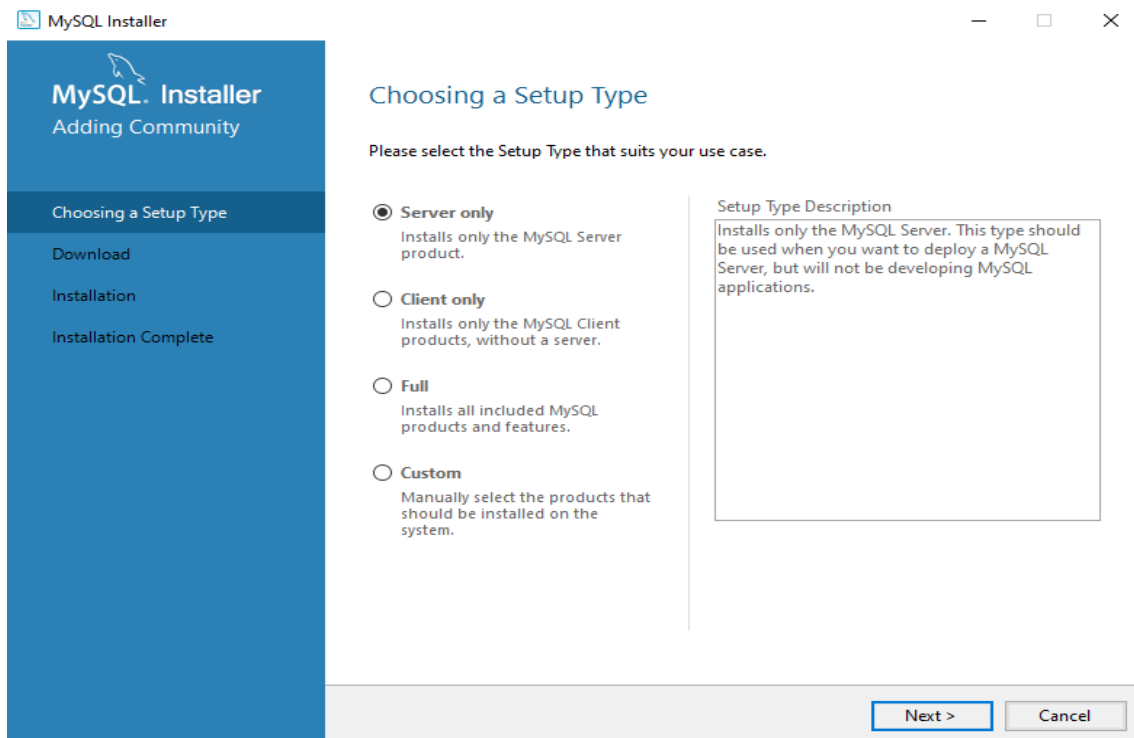


Имя	Описание	Состояние	Тип запуска	Вход от имени
GameSDK Service		Выполняется	Автоматический	Локальная система
Google Chrome Elevation Service			Вручную	Локальная система
GraphicsPerfSvc	Graphics performance ...		Вручную (активный)	Локальная система
KtmRm для координатора ...	Координирует транзакции		Вручную (активный)	Сетевая служба
McpManagementService	<Не удается прочитать...		Вручную	Локальная система
MessagingService_389c7	Служба, отвечающая за...		Вручную (активный)	Локальная система
Microsoft App-V Client	Manages App-V users a...		Отключена	Локальная система
Microsoft Edge Elevation Service	Обеспечивает обновле...		Вручную	Локальная система
Microsoft Edge Update Service	Обеспечивает актуаль...		Автоматический	Локальная система
Microsoft Edge Update Service	Обеспечивает актуаль...		Вручную (активный)	Локальная система
Microsoft Update Health Service	Maintains Update Health		Отключена	Локальная система
MySQL83		Выполняется	Автоматический	Сетевая служба
NPSMSvc_389c7	<Не удается прочитать...	Выполняется	Вручную	Локальная система
OneSyncSvc_389c7	Эта служба синхрониз...	Выполняется	Автоматический	Локальная система
OpenSSH Authentication Agent	Agent to hold private ke...		Вручную	Локальная система
OpenVPN Service			Вручную	Локальная система
P9RdrService_389c7	Включает файловые с...		Вручную (активный)	Локальная система
Peer Networking Identity Manager	Предоставляет служб...		Вручную	Локальная служба
PenService_389c7	Служба пера		Вручную (активный)	Локальная система
PimIndexMaintenanceSvc_...	Индексирует контактные...	Выполняется	Вручную	Локальная система
Plug and Play	Позволяет компьютеру...	Выполняется	Вручную	Локальная система
postgresql-x64-16	Provides relational data...	Выполняется	Автоматический	Сетевая служба
PrintWorkflowUserSvc_389c7	Обеспечивает поддер...		Вручную (активный)	Локальная система

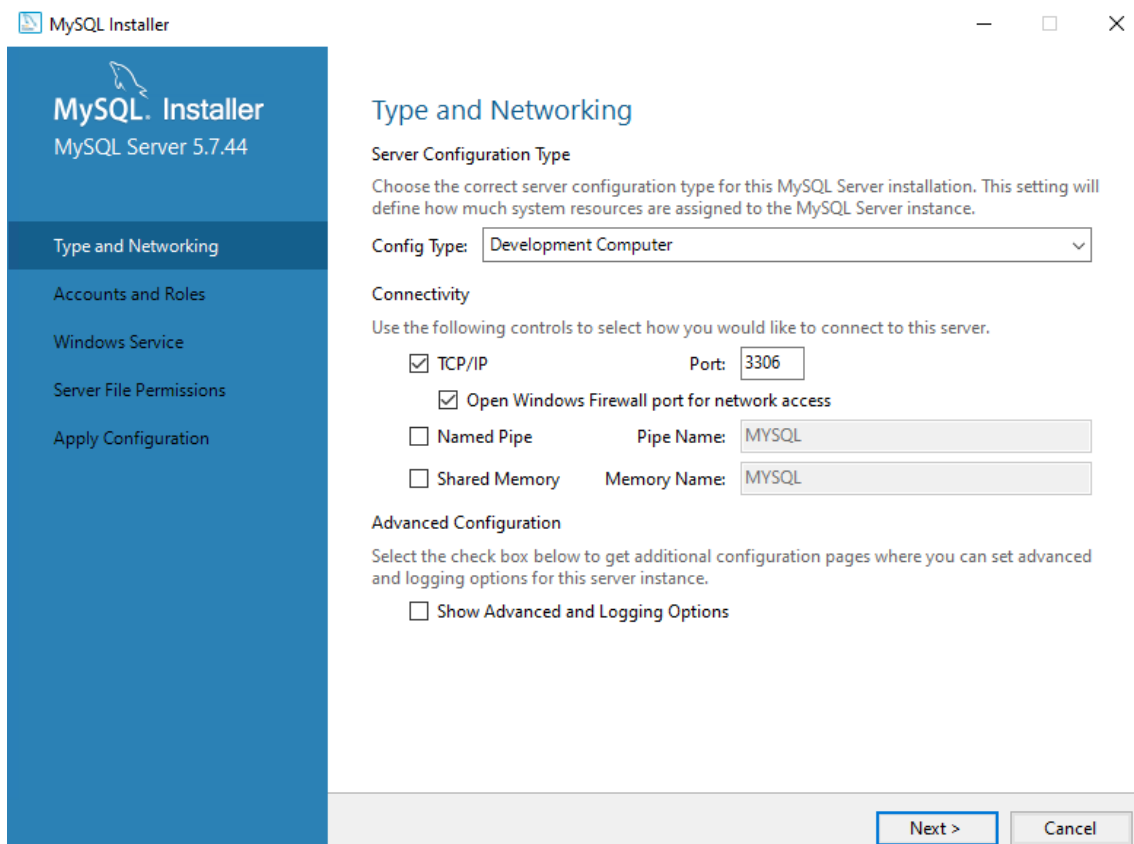
Интересующая служба будет называться **MySQL<версия MySQL>**, а состояние должно быть **Выполняется**.

Установка СУБД MySQL Версии 5.7.44

Установка СУБД MySQL версии 5.7.44 немного отличается от версии 8.3.



На этом этапе выбираем **Server only** и нажимаем **Next >**.



На этом этапе указываем порт (поле **Port**), который будет прослушивать СУБД и нажимаем **Next >**.

MySQL Installer

MySQL Server 5.7.44

Type and Networking

Accounts and Roles

Windows Service

Server File Permissions

Apply Configuration

Accounts and Roles

Root Account Password
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password strength: **Weak**

MySQL User Accounts
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role

Add User

Edit User

Delete

< Back Next > Cancel

Тут указываем пароль суперпользователя и нажимаем **Next >**.

MySQL Installer

MySQL Server 5.7.44

Type and Networking

Accounts and Roles

Windows Service

Server File Permissions

Apply Configuration

Windows Service

☒ Configure MySQL Server as a Windows Service

Windows Service Details
Please specify a Windows Service name to be used for this MySQL Server instance. A unique name is required for each instance.

Windows Service Name:

☒ Start the MySQL Server at System Startup

Run Windows Service as ...
The MySQL Server needs to run under a given user account. Based on the security requirements of your system you need to pick one of the options below.

☒ **Standard System Account**
Recommended for most scenarios.

☐ **Custom User**
An existing user account can be selected for advanced scenarios.

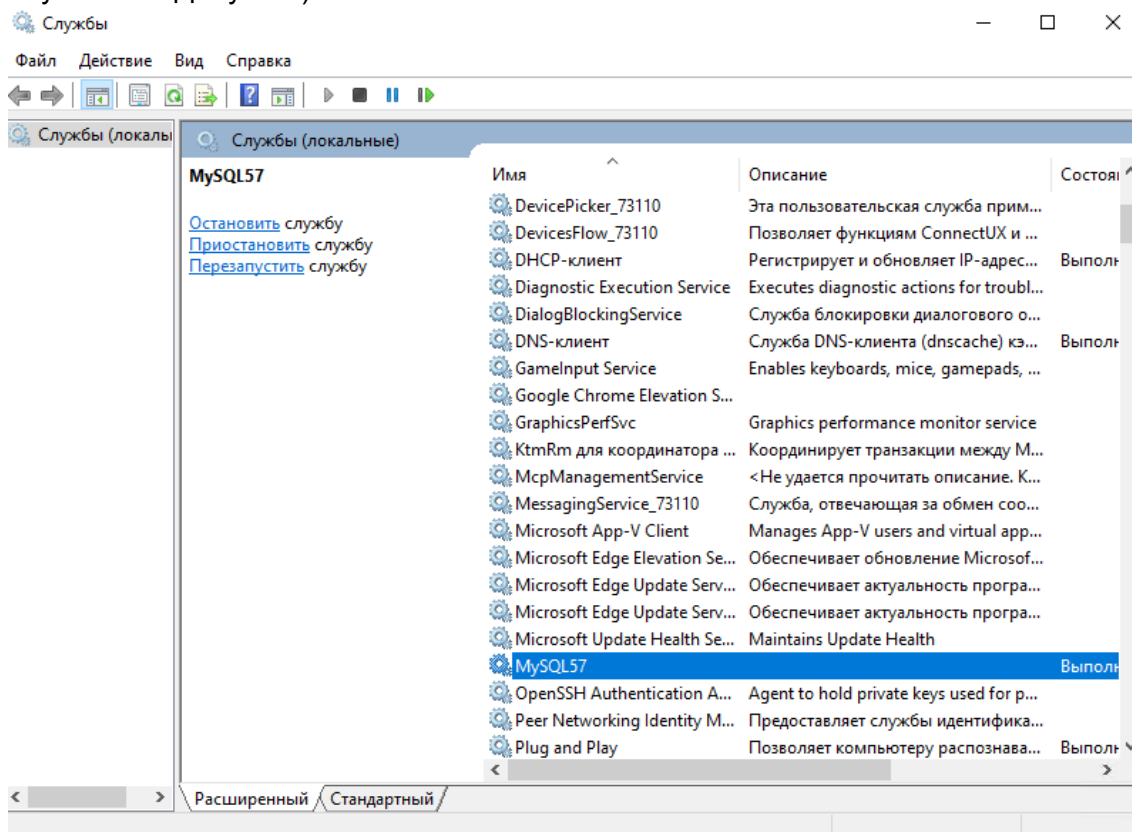
< Back Next > Cancel

Тут оставляем все как есть и нажимаем **Next >**.

Далее соглашаемся со всем пока СУБД не будет установлена.

Для проверки правильности установки надо/можно проверить список служб Windows (если все сделано по инструкции, то должна была создаваться

служба СУБД MySQL):



Интересующая служба будет называться **MySQL<версия MySQL>**, а состояние должно быть **Выполняется**.

Создание БД для контроля КМ

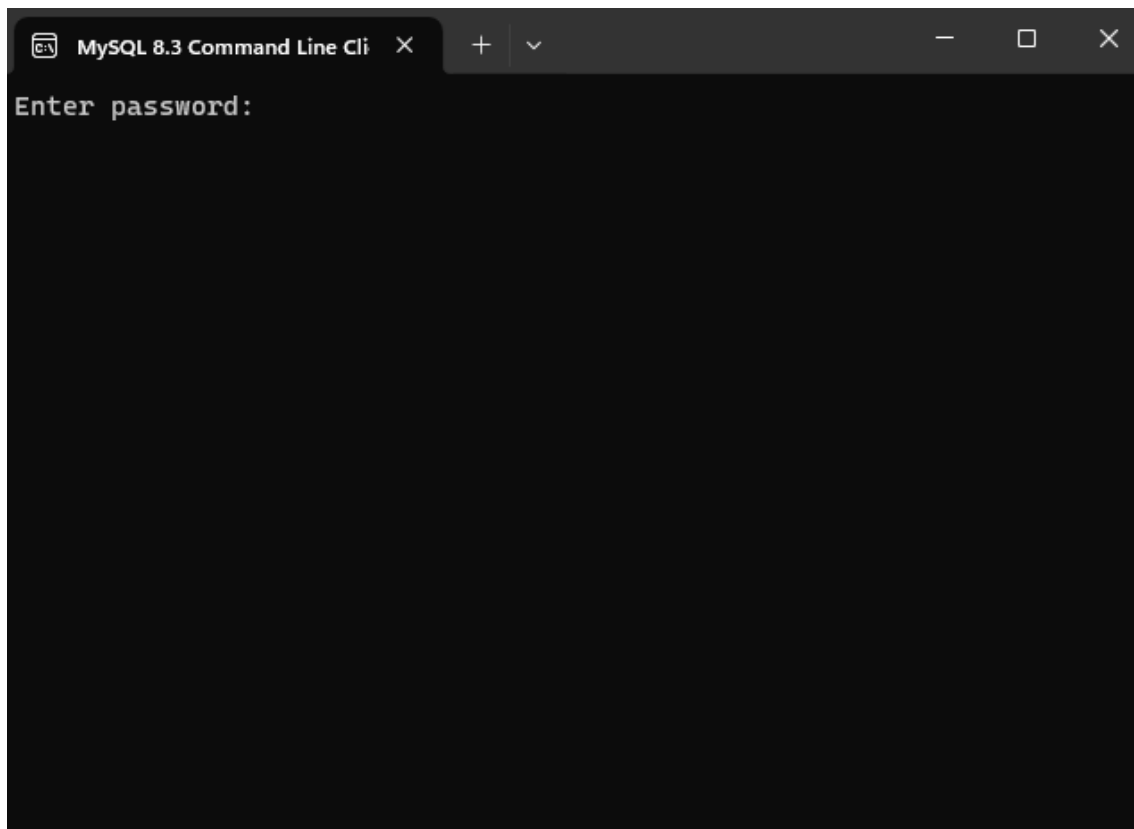
Создать БД для контроля КМ надо воспользоваться утилитой **MySQL Command Line Client**.

Запуск утилиты **MySQL Command Line Client**

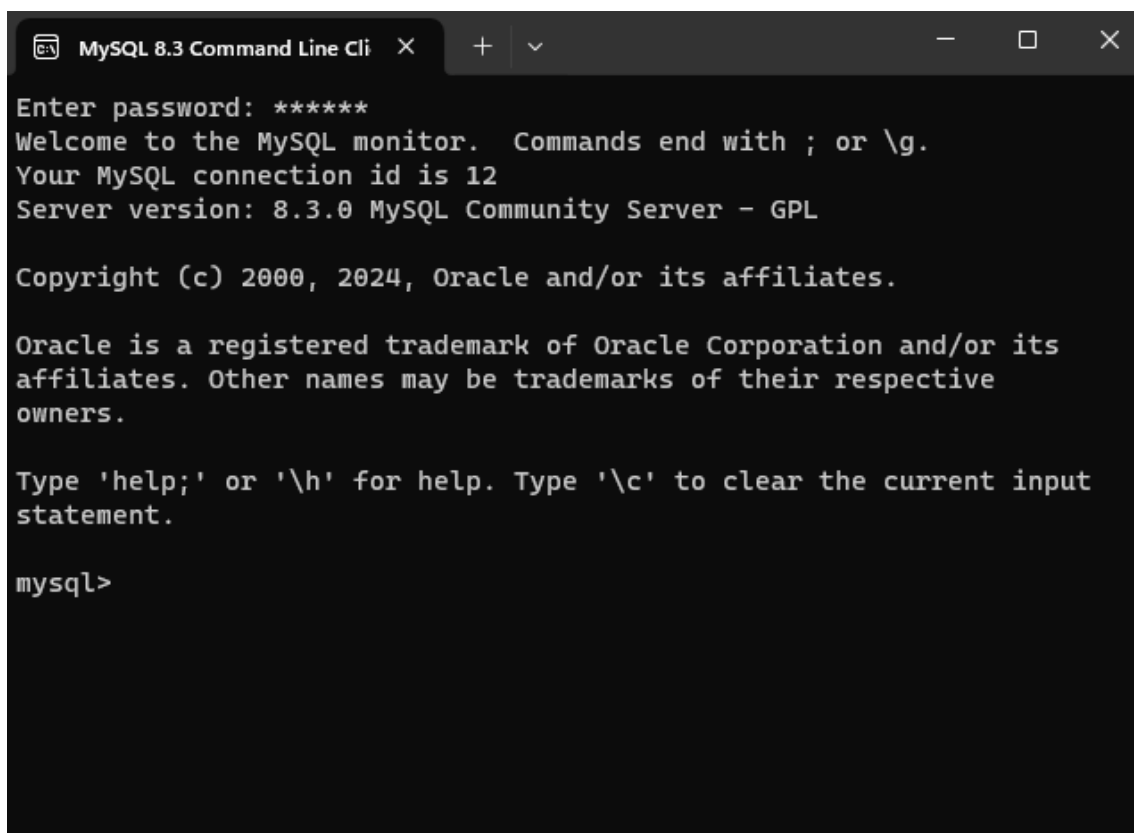
В данной инструкции будет рассмотрен вариант создания БД средствами утилиты **MySQL 8.3 Command Line Client**, которая была установлена вместе с сервером.

Для запуска утилиты **MySQL Command Line Client** необходимо:

1. Открыть меню **ПУСК**
2. Ввести **MySQL** и в подобранном списке выбрать утилиту с наименованием **MySQL <номер версии> Command Line Client** или вручную найти в списке программ
3. Запустить утилиту



Вводим пароль пользователя суперпользователя, указанный при конфигурировании СУБД (сразу после установки) и жмем Enter.



Если введенный пароль верен, то подключение устанавливается, а вывод с командной строке становится похожим на то, что на скриншоте.

Если все получилось, то можно последняя строка будет **mysql>**, что является приглашением для ввода запроса и в таком случае можно переходить к следующему этапу.

Создание БД для контроля КМ

Если по окончании предыдущего этапа последняя строка в консоли выглядит так - **mysql>** (это приглашение для ввода пароля), то мы можем ввести текст запроса, который будет выполнен после нажатия **Enter**.

Для создания БД надо выполнить следующий запрос (обратите внимание, что это все одна строка) введя его, после чего нажать **Enter**:

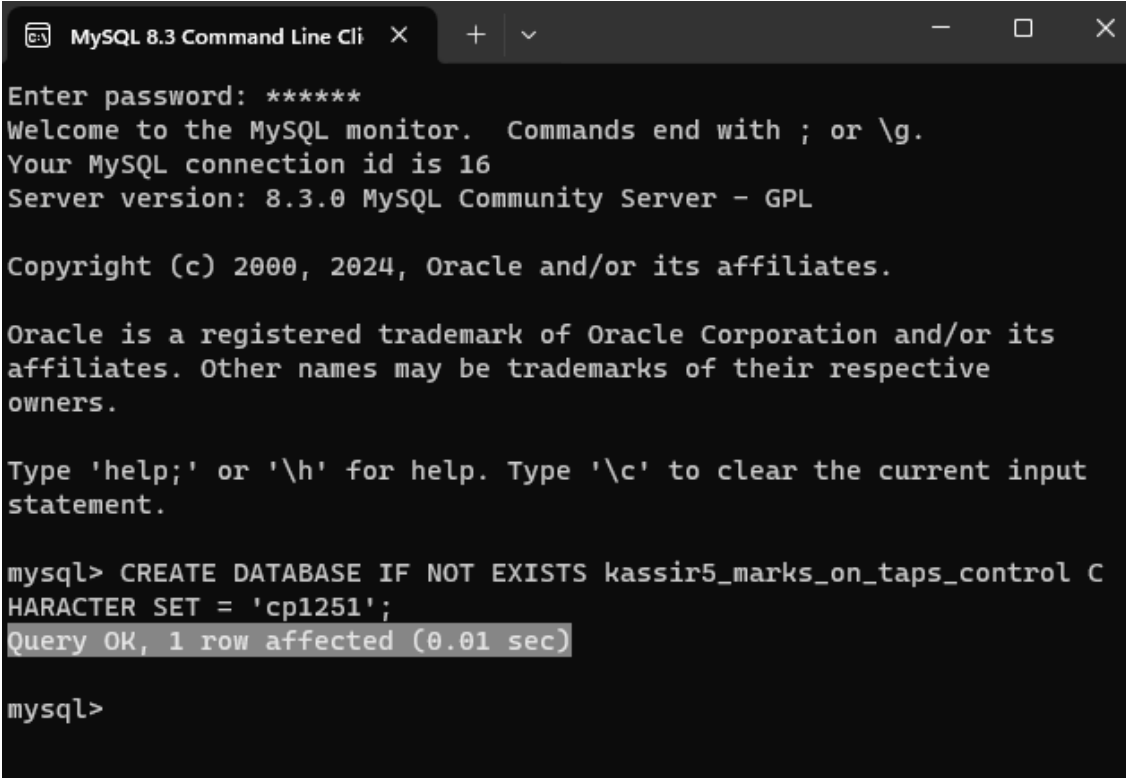
CREATE DATABASE IF NOT EXISTS <Имя базы данных> CHARACTER SET = 'cp1251';

Обратите внимание!

<Имя базы данных> - это строка состоящая из букв латинского алфавита, цифр и символа Нижнее подчеркивание, БЕЗ пробелов и без кавычек. Как на скриншоте далее.

При этом всё, начинаться имя должно с буквы.

Рекомендуется следующее наименование: **kassir5_marks_on_taps_control**.



```
MySQL 8.3 Command Line Cli X + - □ X
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> CREATE DATABASE IF NOT EXISTS kassir5_marks_on_taps_control C
HARACTER SET = 'cp1251';
Query OK, 1 row affected (0.01 sec)

mysql>
```

Если все сделано верно, что в итоге в консоль выведется строка **Query OK, 1 row affected (... sec)**. На этом с настройками СУБД MySQL закончили.

Можно переходить к настройке ПК с ПО “Кассир 5”.

Установка и настройка СУБД PostgreSQL

Установка СУБД PostgreSQL

Если есть опыт в установке и настройке PostgreSQL, то этот раздел будет мало полезен, так как в установке PostgreSQL для работы контроля КМ нет ничего специфического.

Для начала надо получить установщик PostgreSQL. Для этого переходим посылке: <https://www.postgresql.org/download> или <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>.

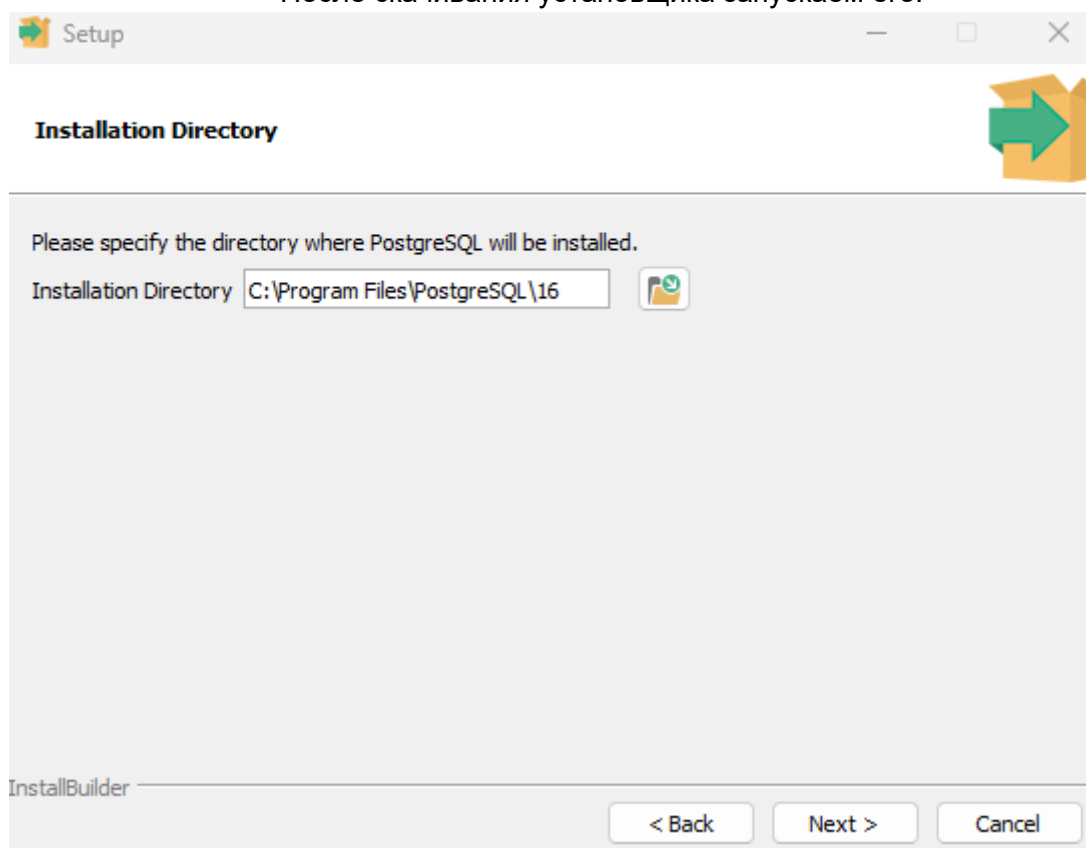
Обратите внимание!

При выборе дистрибутива (установщика) PostgreSQL уделите внимание выбору установщика с учетом разрядности ОС, в которую будет устанавливаться PostgreSQL.

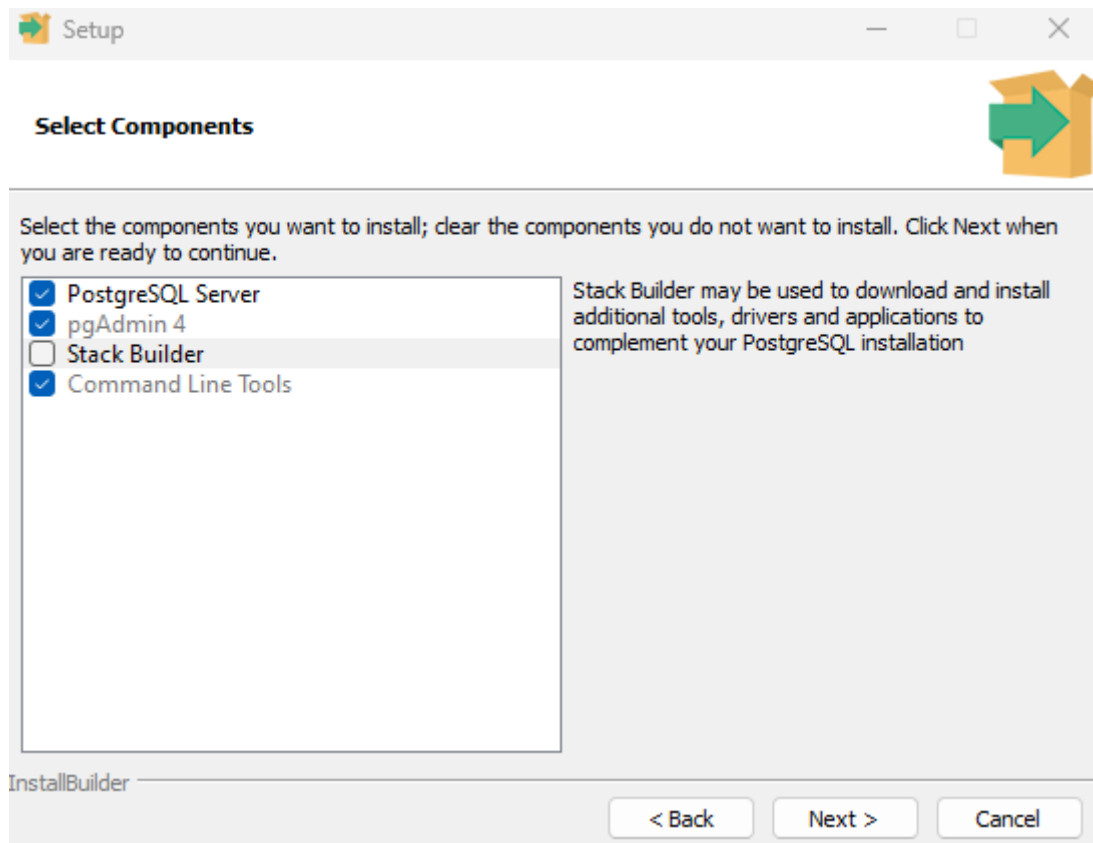
Также обратите внимание, что начиная с версии **11.22** PostgreSQL 32-битные ОС семейства Windows не поддерживаются.

Далее описание процесса установки PostgreSQL будет сопровождаться скриншотами, актуальными для PostgreSQL 16.1 на примере установки в Windows 11.

После скачивания установщика запускаем его.



На данном этапе указываем директорию для установки PostgreSQL и нажимаем **Next >**.

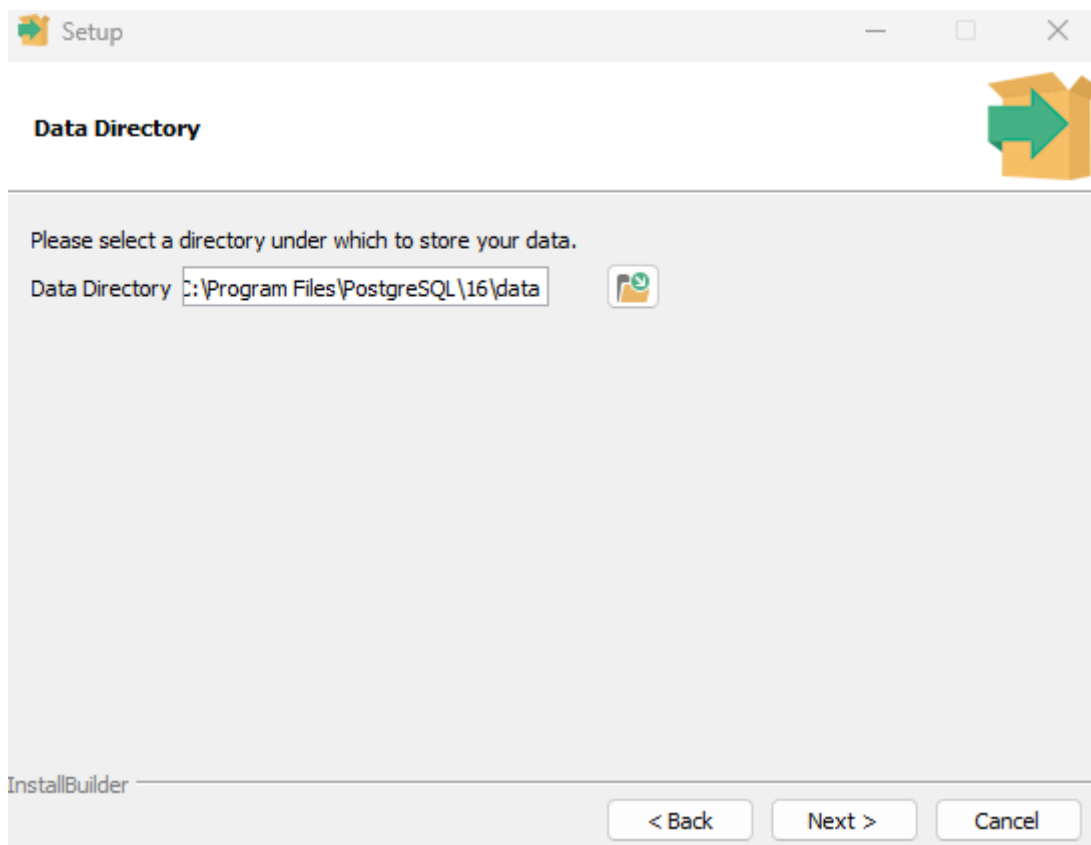


На данном этапе отмечаем компоненты, которые хотим установить.

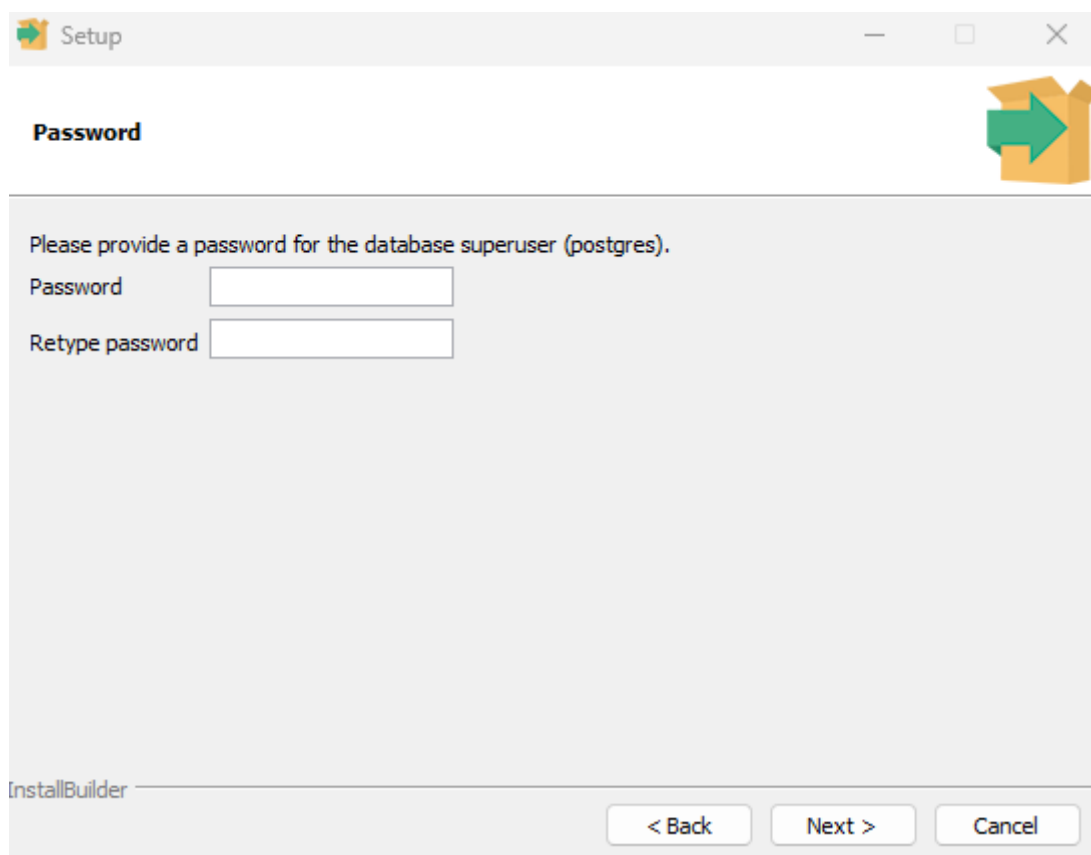
Рекомендуется выбрать те же компоненты, что и на скриншоте выше:

1. PostgreSQL Server (обязательно) - собственно и есть сервер PostgreSQL
2. pgAdmin (необязательно) - графический интерфейс для администрирования PostgreSQL.
3. Stack Builder (необязательно) - это утилита для установки дополнительных инструментов, драйверов и приложений для работы с PostgreSQL.
4. Command Line Tools (обязательно) - инструменты для работы с PostgreSQL посредством консоли (командной строки).


Если выбрали нужные компоненты, то нажимаем **Next >**.



На данном этапе необходимо указать путь до директории, где будут храниться данные БД и нажимаем **Next >**.



На данном этапе нужно дважды указать пароль для суперпользователя (имя: postgres) и нажимаем **Next >**.



Setup


Port

Please select the port number the server should listen on.


Port

InstallBuilder

< Back Next > Cancel



На данном этапе указываем номер свободного порта, который будет прослушиваться в ожидании запросов на подключение. Рекомендуется оставить по умолчанию. Нажимаем **Next >**.



Setup


Advanced Options

Select the locale to be used by the new database cluster.

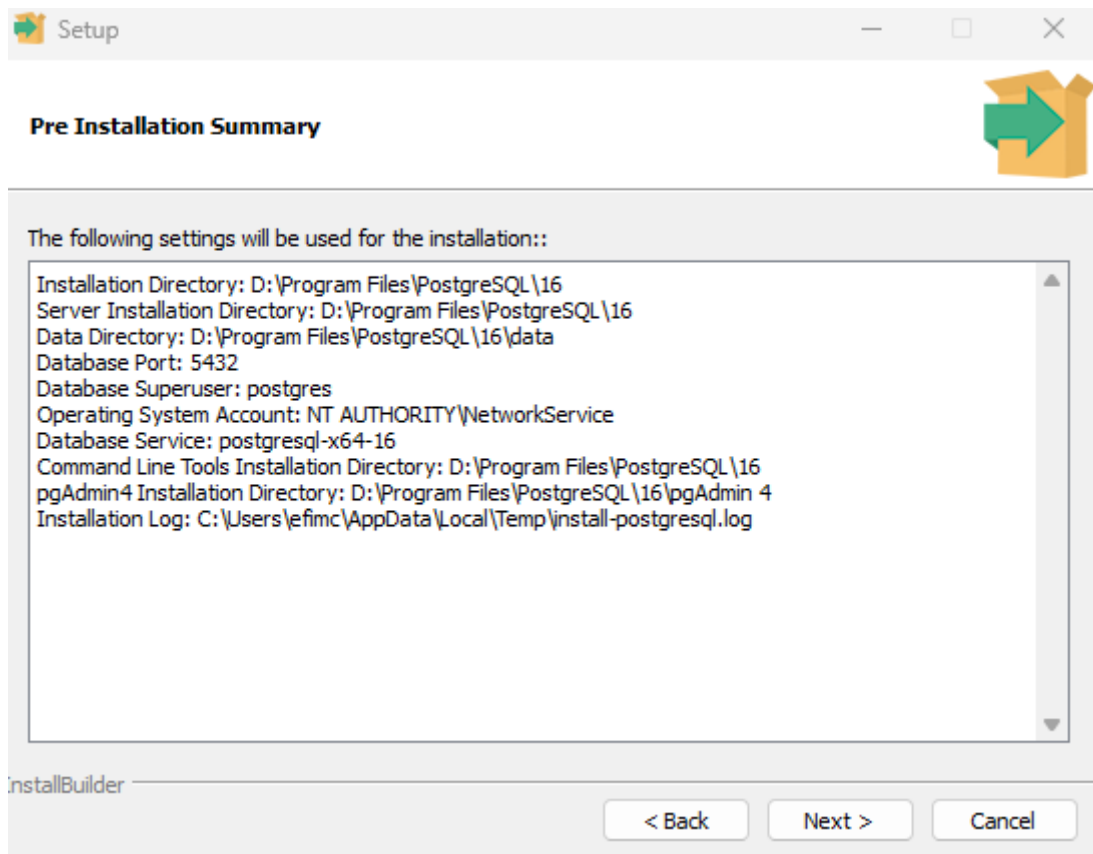
Locale

InstallBuilder

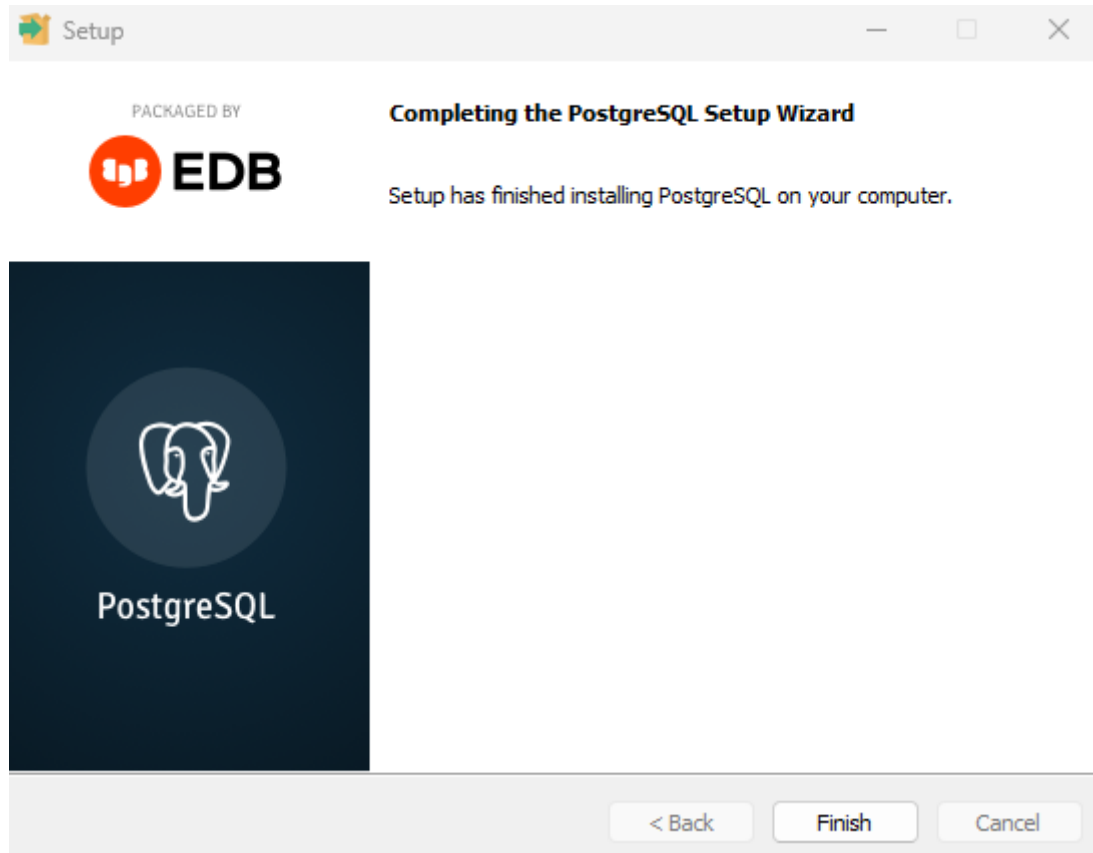
< Back Next > Cancel



На этом этапе указываем региональные параметры (**Russian, Russia**). Нажимаем **Next >**.

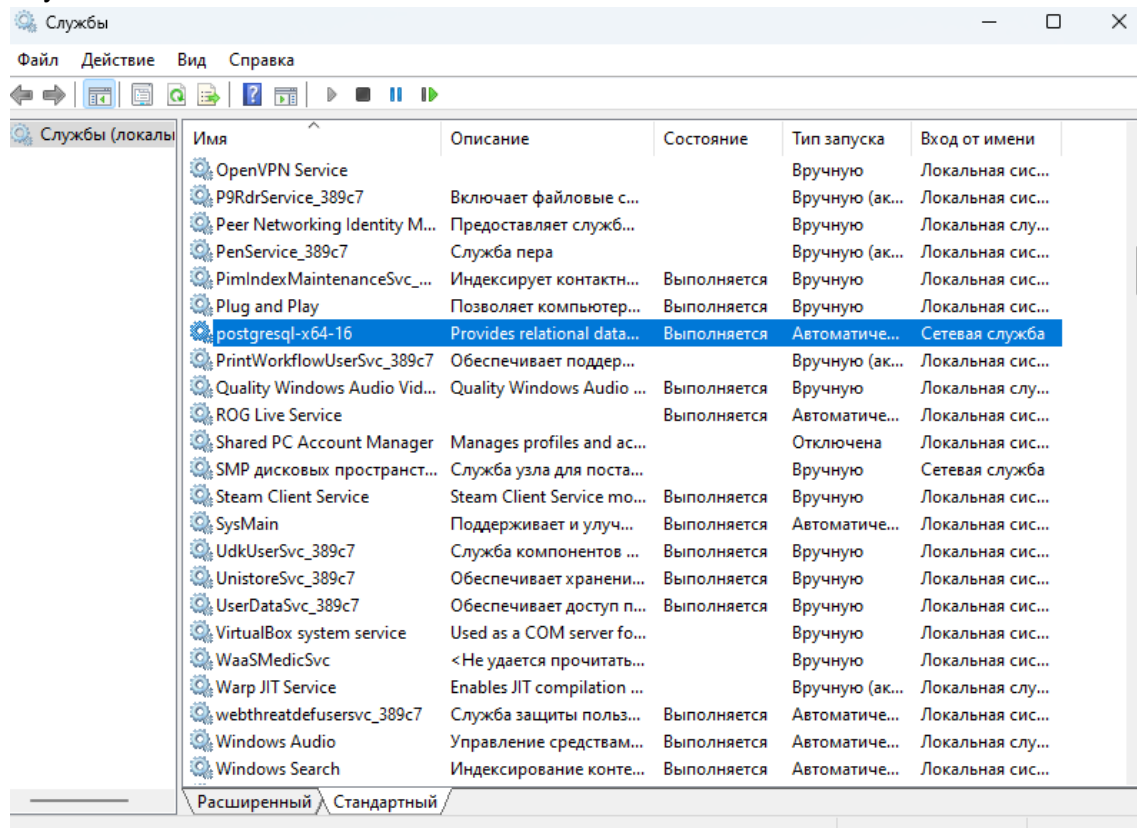


На этом этапе ознакомливаемся с параметрами установки (параметры, компоненты и куда они устанавливаются). Нажимаем **Next >**.



Поздравляем, установка завершена. Нажимаем **Finish**.

Для проверки правильности установки надо/можно проверить список служб Windows.:



Интересующая служба будет называться **postgresql-<обозначение разрядности>-<версия PostgreSQL>**, а состояние должно быть **Выполняется**.

Создание БД для контроля КМ

Создать БД для контроля КМ можно двумя способами, который различаются лишь удобством взаимодействия: средствами утилиты **SQL Shell (psql)** или при помощи pgAdmin.

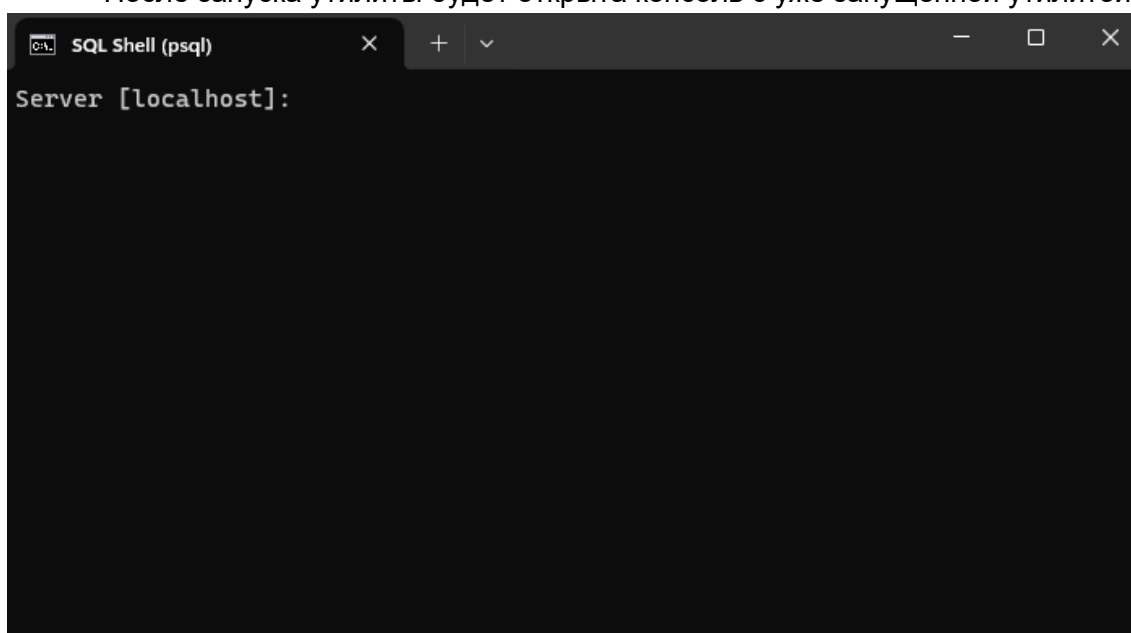
Запуск утилиты **SQL Shell (psql)**

В данной инструкции будет рассмотрен вариант создания БД средствами утилиты **SQL Shell (psql)**, которая была установлена вместе с сервером.

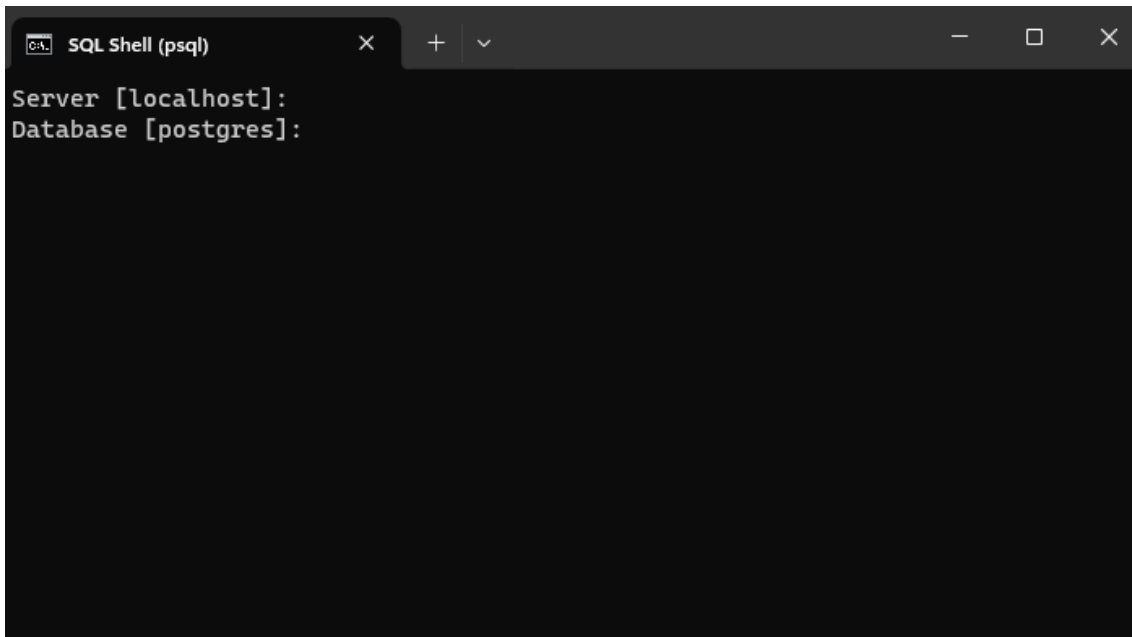
Для запуска утилиты **SQL Shell (psql)** необходимо:

4. Открыть меню **ПУСК**
5. Ввести **SQL Shell (psql)** или вручную найти в списке программ
6. Запустить утилиту

После запуска утилиты будет открыта консоль с уже запущенной утилитой:

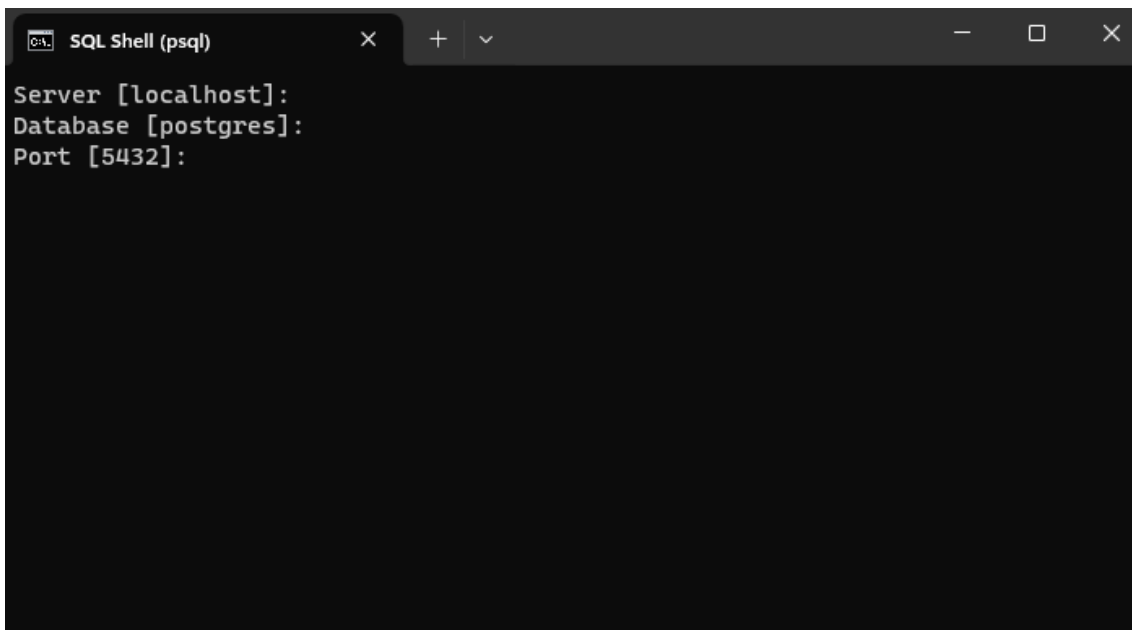


Сначала надо указать ip-адрес СУБД (**Server**). если утилита запускается на той же машине, на которой установлена СУБД, то можно просто нажать **Enter**, пропустив ввод, так как адрес localhost уже указан, как значение по умолчанию (в квадратных скобках) для данного параметра.

A screenshot of a terminal window titled "SQL Shell (psql)". The window has a dark background and a light-colored title bar with standard window controls (close, maximize, and a dropdown menu). The terminal text shows the connection prompt: "Server [localhost]:" followed by "Database [postgres]:".

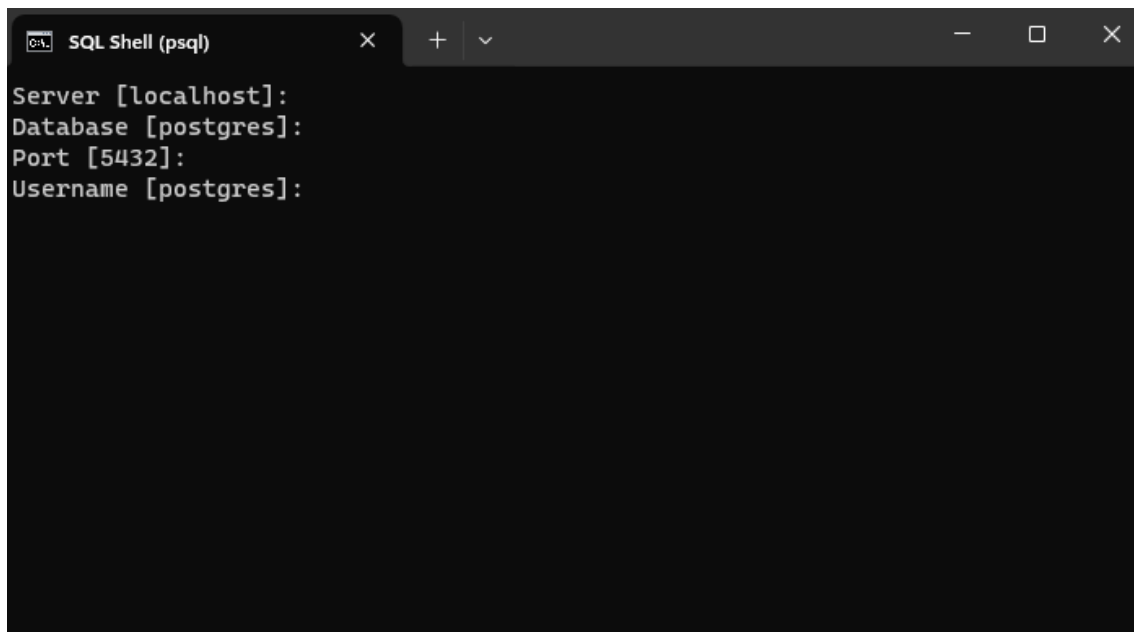
```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
```

На этом этапе надо было бы указать имя БД (**Database**), к которой мы подключаемся, но так как мы только собираемся создать БД, то можем подключиться к БД **postgres**, которая создается автоматически при установке СУБД PostgreSQL. Ничего не печатая, снова нажимаем Enter, так как БД **postgres** указана по умолчанию (в квадратных скобках).

A screenshot of a terminal window titled "SQL Shell (psql)". The window has a dark background and a light-colored title bar with standard window controls. The terminal text shows the connection prompt: "Server [localhost]:", "Database [postgres]:", and "Port [5432]:".

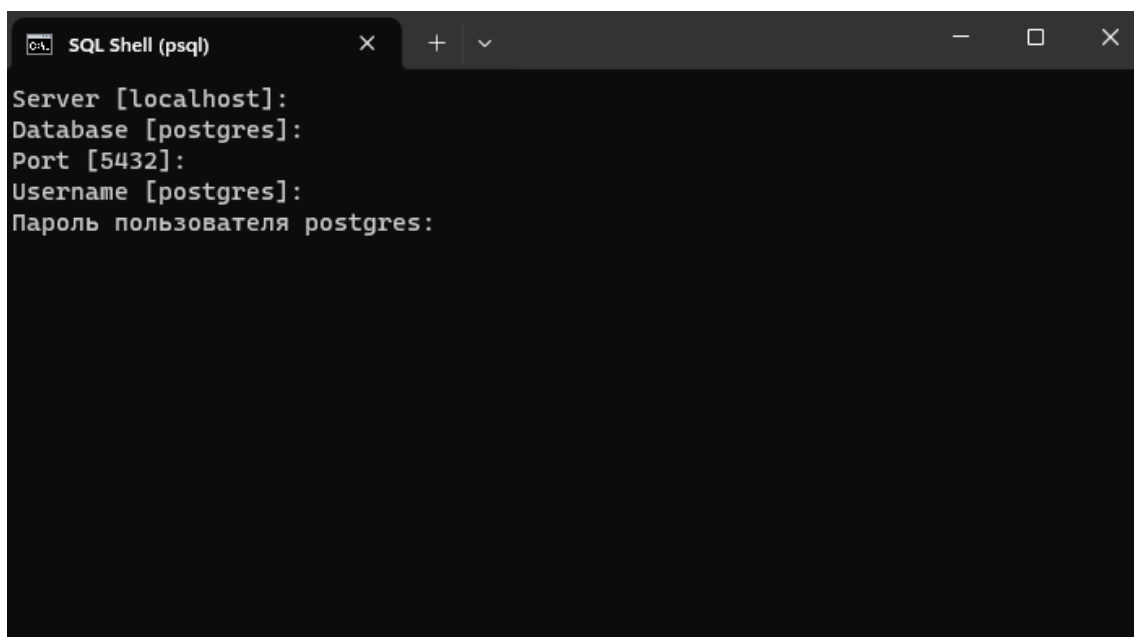
```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
```

На этом этапе надо указать порт, который прослушивает СУБД. если при установке был указан порт по умолчанию (5432), то можно пропустить этот этап нажатием **Enter**, так как и тут этот же порт указан по умолчанию.



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
```

Тут указываем имя пользователя, под которым хотим подключиться к СУБД. По умолчанию это postgres. Оставляем и нажимаем **Enter**.



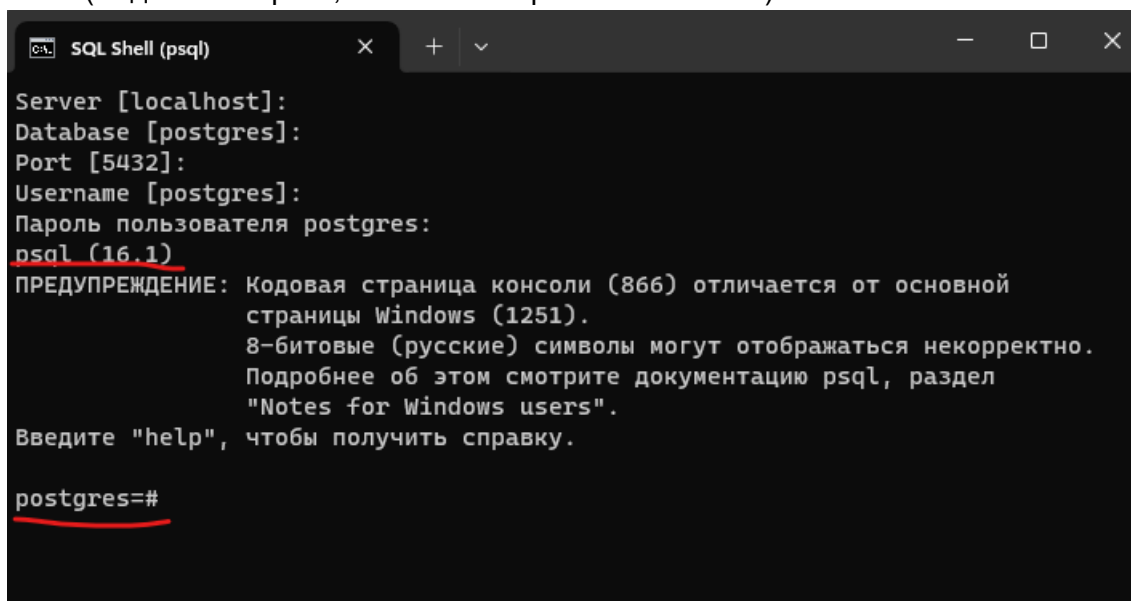
```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Пароль пользователя postgres:
```

На этом этапе отделаться значением по умолчанию уже не получится. Вводим пароль суперпользователя, который указывали при установке сервера, затем нажимаем **Enter**.

Обратите внимание!

При вводе пароля символы в консоли не выводятся.

Если все сделано правильно (то есть данные указаны верно, а служба СУБД запущена), то ваша консоль должна выглядеть примерно так, как на скрине ниже (выделены строки, наличие которых обязательно):



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Пароль пользователя postgres:
psql (16.1)
ПРЕДУПРЕЖДЕНИЕ: Кодовая страница консоли (866) отличается от основной
                  страницы Windows (1251).
                  8-битовые (русские) символы могут отображаться некорректно.
                  Подробнее об этом смотрите документацию psql, раздел
                  "Notes for Windows users".
Введите "help", чтобы получить справку.
postgres=#
```

Если ваша консоль выглядит иначе и вместо строки **postgres=#** вы видите что-то похожее на сообщение об ошибке, то попробуйте еще раз запустить утилиту и ввести все заново, а также проверьте запущена ли соответствующая служба СУБД PostgreSQL.

Если же все получилось, то можно переходить к следующему этапу.

Создание БД для контроля КМ

Если по окончании предыдущего этапа последняя строка в консоли выглядит так - **postgres=#** (это приглашение для ввода пароля), то мы можем ввести текст запроса, который будет выполнен после нажатия **Enter**.

Для создания БД надо выполнить следующий запрос (обратите внимание, что это все одна строка):

```
CREATE DATABASE <Имя базы данных> WITH OWNER = postgres TEMPLATE =  
template0 ENCODING = 'WIN1251';
```

Обратите внимание!

<Имя базы данных> - это строка состоящая из букв латинского алфавита, цифр и символа **Нижнее подчеркивание**, **БЕЗ** пробелов.

При этом всё, начинаться имя должно с буквы.

Рекомендуется следующее наименование: **kassir5_marks_on_taps_control**.

Если после ввода текста запроса и нажатия Enter будет выведена строка **CREATE DATABASE**, как на скриншоте далее, то можно считать, что БД была успешно создана.


```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Пароль пользователя postgres:
psql (16.1)
ПРЕДУПРЕЖДЕНИЕ: Кодовая страница консоли (866) отличается от основной
                  страницы Windows (1251).
                  8-битовые (русские) символы могут отображаться некорректно.
                  Подробнее об этом смотрите документацию psql, раздел
                  "Notes for Windows users".
Введите "help", чтобы получить справку.

postgres=# CREATE DATABASE kassir5_marks_on_taps_control WITH OWNER = postgr
es TEMPLATE = template0 ENCODING = 'WIN1251';
CREATE DATABASE
postgres=#
```

Далее нужно создать таблицы в базе данных.

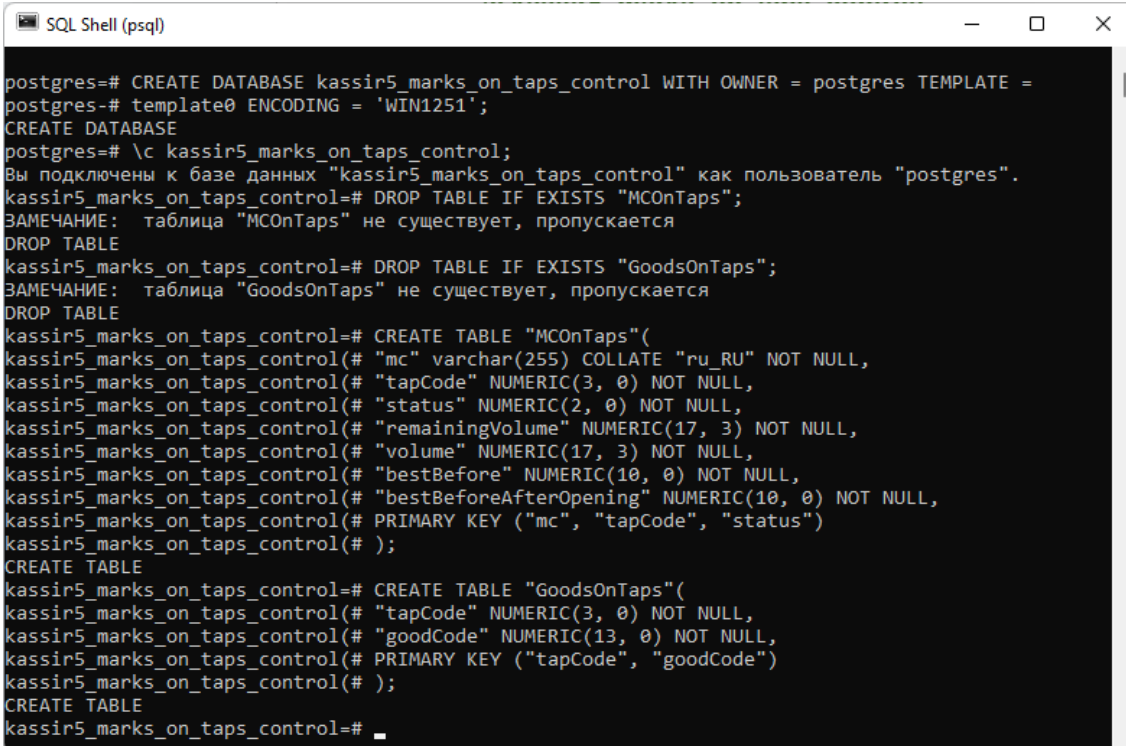
Обратите внимание!

В релизе 5.1.7.19 таблицы в PostgreSQL нужно создавать в СУБД, а не через Кассир 5.

Для создания БД надо выполнить следующий запрос:

```
\c kassir5_marks_on_taps_control;
DROP TABLE IF EXISTS "MConTaps";
DROP TABLE IF EXISTS "GoodsOnTaps";
CREATE TABLE "MConTaps"(
    "mc" varchar(255) NOT NULL,
    "tapCode" NUMERIC(3, 0) NOT NULL,
    "status" NUMERIC(2, 0) NOT NULL,
    "remainingVolume" NUMERIC(17, 3) NOT NULL,
    "volume" NUMERIC(17, 3) NOT NULL,
    "bestBefore" NUMERIC(10, 0) NOT NULL,
    "bestBeforeAfterOpening" NUMERIC(10, 0) NOT NULL,
    PRIMARY KEY ("mc", "tapCode", "status")
);
CREATE TABLE "GoodsOnTaps"(
    "tapCode" NUMERIC(3, 0) NOT NULL,
    "goodCode" NUMERIC(13, 0) NOT NULL,
    PRIMARY KEY ("tapCode", "goodCode")
);
```

Ожидаемый результат:



```
SQL Shell (psql)

postgres=# CREATE DATABASE kassir5_marks_on_taps_control WITH OWNER = postgres TEMPLATE =
postgres=# template0 ENCODING = 'WIN1251';
CREATE DATABASE
postgres=# \c kassir5_marks_on_taps_control;
Вы подключены к базе данных "kassir5_marks_on_taps_control" как пользователь "postgres".
kassir5_marks_on_taps_control=# DROP TABLE IF EXISTS "MConTaps";
ЗАМЕЧАНИЕ: таблица "MConTaps" не существует, пропускается
DROP TABLE
kassir5_marks_on_taps_control=# DROP TABLE IF EXISTS "GoodsOnTaps";
ЗАМЕЧАНИЕ: таблица "GoodsOnTaps" не существует, пропускается
DROP TABLE
kassir5_marks_on_taps_control=# CREATE TABLE "MConTaps"(
kassir5_marks_on_taps_control(# "mc" varchar(255) COLLATE "ru_RU" NOT NULL,
kassir5_marks_on_taps_control(# "tapCode" NUMERIC(3, 0) NOT NULL,
kassir5_marks_on_taps_control(# "status" NUMERIC(2, 0) NOT NULL,
kassir5_marks_on_taps_control(# "remainingVolume" NUMERIC(17, 3) NOT NULL,
kassir5_marks_on_taps_control(# "volume" NUMERIC(17, 3) NOT NULL,
kassir5_marks_on_taps_control(# "bestBefore" NUMERIC(10, 0) NOT NULL,
kassir5_marks_on_taps_control(# "bestBeforeAfterOpening" NUMERIC(10, 0) NOT NULL,
kassir5_marks_on_taps_control(# PRIMARY KEY ("mc", "tapCode", "status")
kassir5_marks_on_taps_control(# );
CREATE TABLE
kassir5_marks_on_taps_control=# CREATE TABLE "GoodsOnTaps"(
kassir5_marks_on_taps_control(# "tapCode" NUMERIC(3, 0) NOT NULL,
kassir5_marks_on_taps_control(# "goodCode" NUMERIC(13, 0) NOT NULL,
kassir5_marks_on_taps_control(# PRIMARY KEY ("tapCode", "goodCode")
kassir5_marks_on_taps_control(# );
CREATE TABLE
kassir5_marks_on_taps_control=#
```

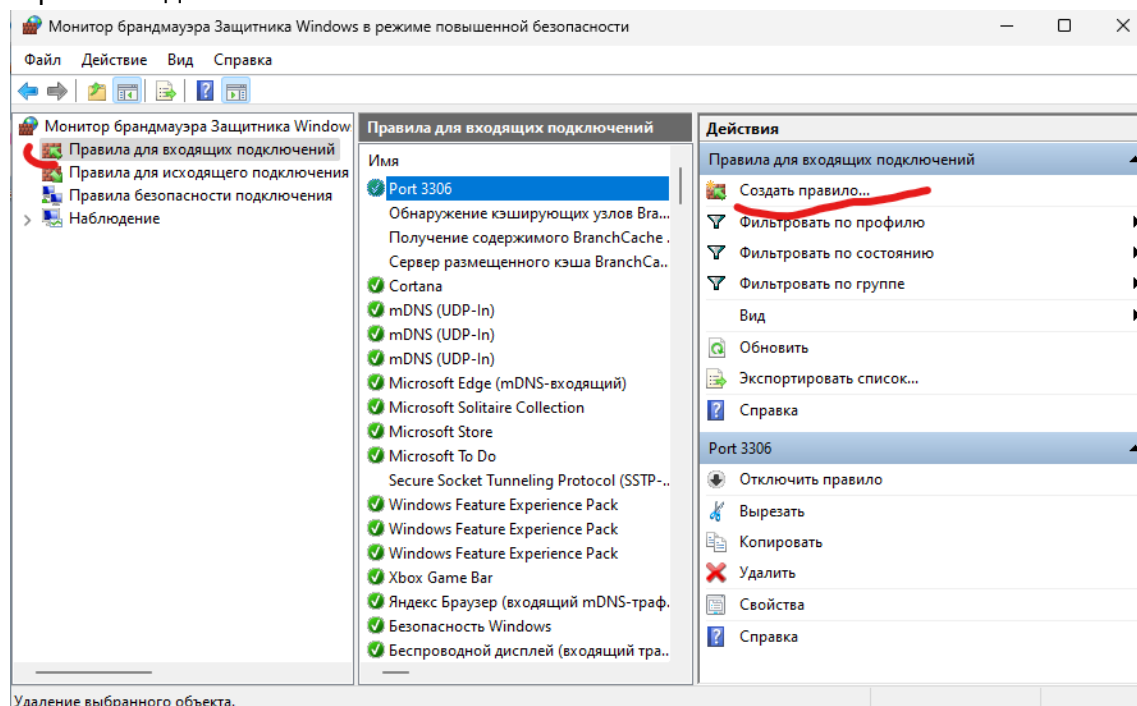
Если БД и таблицы были созданы, то с настройками СУБД PostgreSQL закончили. Можно переходить к настройке ПК с ПО “Кассир 5”.

Дополнительные настройки для удаленного доступа к SQL-серверу

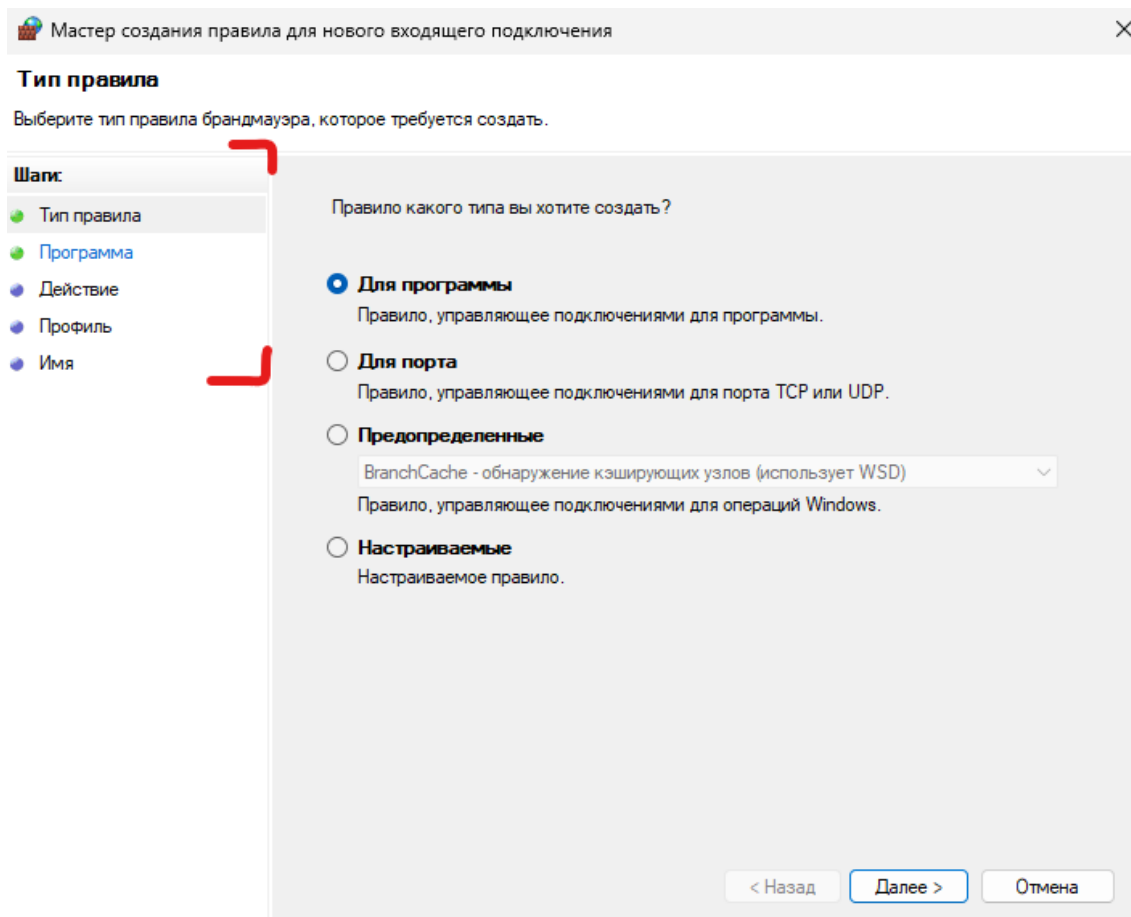
Настройка правил брандмауэра

Первым делом, в настройках брандмауэра Windows нужно открыть доступ для входящих подключений к прослушиваемому SQL-сервером порту.

Для этого необходимо открыть программу **Монитор брандмауэра Защитника Windows в режиме повышенной безопасности** (Меню Пуск - вводим строку “Монитор брандмауэра” - в предложенном списке выбираем соответствующую программу). После чего должна открыться форма, как на скриншоте далее:



В списке слева переходим в ветку **Правила для входящих подключений** (выделено на скриншоте), далее в меню **Действия** нажимаем “Создать правило...” (так же выделено на скриншоте), после чего откроется **Мастер создания правила для нового входящего подключения** (см. скриншот далее).



Тут и далее для навигации по этапам настройки правила будут использоваться наименования **Шагов** (выделено на скриншоте).

На этапе **Тип правила** выбираем пункт **Для порта**. Затем жмем кнопку **“Далее >”**.

На этапе **Протокол и порты** (на скриншоте этот этап называется **Программа**, так как имя этого этапа зависит от выбора сделанного на этапе **Тип правила**):

1. Указываем, что протокол, к которому будет применяться правило - это **Протокол TCP**
2. Указываем, что это правило будет применяться к определенному порту, выбирая пункт **“Определенные локальные порты:”** и в соответствующем поле ввода указываем номер порта, который прослушивается ранее настроенным SQL-сервером (напоминаю, что по умолчанию для MySQL - это 3306, а для PostgreSQL - 5432. Если при установке и настройке SQL-сервера номер порта не изменялся, то используем соответствующее типу СУБД значение номера порта по умолчанию, иначе вводим тот же номер порта, который был указан при настройке)
3. Жмем кнопку **“Далее >”**.

На этапе **Действие** выбираем пункт **Разрешить подключение**. Затем жмем кнопку **“Далее >”**.

На этапе **Профиль** советуется поставить **ВСЕ** галочки, если вы не знаете или не понимаете, как настроена ваша локальная сеть. Затем жмем кнопку **“Далее >”**.

На этапе **Имя** указываем имя и описание. Указывать описание необязательно, но рекомендуется для того, чтобы в будущем иметь понимание о том, кто и зачем создал это правило. Затем жмем кнопку **Готово**.

Настройка SQL-сервера для прослушивания внешних подключений

По умолчанию SQL-сервер может быть настроен на подключения только с локальной машины, на которой установлен сервер.

Для настройки прослушивания подключений с внешних рабочих машин надо отредактировать соответствующие конфигурационные файлы и после сохранения изменений перезагрузить SQL-сервер (службу сервера).

Обратите внимание!

После редактировании любых конфигурационных файлов сервера необходимо перезапустить этот сервер, иначе изменения не будут применены.

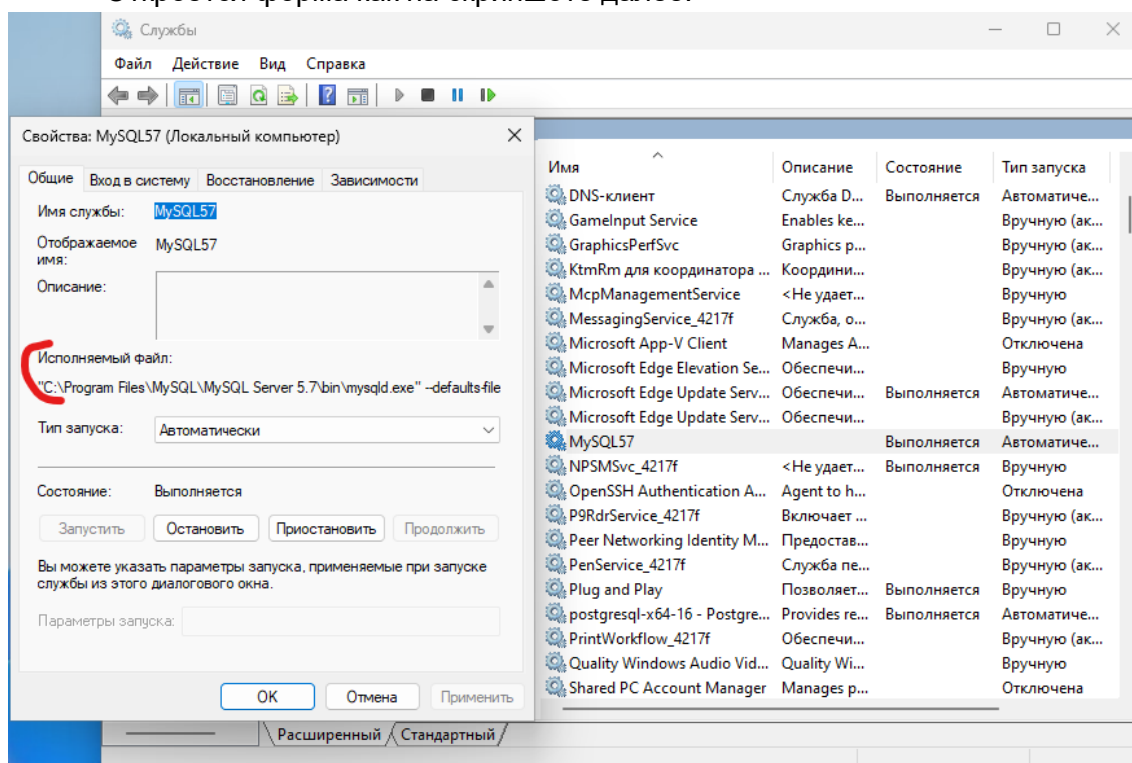
MySQL

Для того, чтобы MySQL прослушивал подключение с внешних рабочих машин, нужно прописать параметр **bind-address** в конфигурационном файле сервера, который находится... да он где угодно может находиться.

Проще всего узнать расположение конфигурационного файла, который использует ваш сервер - это:

1. открыть список служб Windows, найти соответствующую службу сервера MySQL
2. кликнуть на найденную службу в списке правой клавишей мыши
3. в появившемся списке выбрать пункт **Свойства**

Откроется форма как на скриншоте далее:



В этой форме нас интересует свойство **Исполняемый файл**, в котором будет значиться строка подобная этой (в зависимости от версии установленного

сервера и места установки выделенные зеленым цветом участки строки могут отличаться):

```
"C:\Program Files\MySQL\MySQL Server 5.7\bin\mysqld.exe" --defaults-file="C:\ProgramData\MySQL\MySQL Server 5.7\my.ini" MySQL57
```

В указанной строке нас интересует выделенная кавычками часть после **--defaults-file=**, в нашем случае это **"C:\ProgramData\MySQL\MySQL Server 5.7\my.ini"**. Это и есть путь до конфигурационного файла, используемого нашим сервером.

После того, как мы установили расположение конфигурационного файла, его можно открыть в любом текстовом редакторе (не редакторе документов, вроде WORD, а текстовым - вроде блокнота).

Обратите внимание!

Если вы не видите на диске **C** папку **ProgramData**, то не стоит паниковать. Она просто скрыта настройками Windows от случайного доступа шаловливых ручек. В таком случае можно отключить скрытие папок или в адресную строку (**Ctrl + L**) **Проводника** ввести путь **C:\ProgramData** и папка откроется.

Сам конфигурационный файл разделен на секции, которые обозначаются следующей строкой: [**наименование секции**]. После обозначения секции уже заполняются параметры соответствующей секции. Каждая секция отвечает за отдельные программные модули, например, [**mysqld**] - это секция для параметров именно сервера MySQL.

Обратите внимание!

По формату конфигурационного файла строки начинающиеся с символов **#** или **;** (точка с запятой) являются комментариями и игнорируются сервером при установке значений соответствующих параметров.

В конфигурационном файле нас интересует секция [**mysqld**], в котором можно/надо указывать параметр **bind-address**, который в свою очередь определяет список ip-адресов, с которых возможно подключение к серверу. Обратите внимание, что такого параметра может не быть в конфигурационном файле, тогда будет использоваться значение по умолчанию * ('звездочка'), что в свою очередь разрешает любое подключение от внешних хостов. В случае если в конфигурационном файле нет этого параметра, лучше добавить его с той же звездочкой. Если же такой параметр есть в файле и он не закомментирован, то необходимо установить ему значение * ('звездочка') (подробнее о значениях данного параметра можно почитать тут:

https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html#sysvar_bind_address), как на скриншоте далее:

```
# you have installed the server correctly (see above) so to read this
# file.
#
# server_type=2
[mysqld]
# The next three options are mutually exclusive to SERVER_PORT below.
# skip-networking
# enable-named-pipe
# shared-memory

# shared-memory-base-name=MYSQL

# The Pipe the MySQL Server will use
# socket=MYSQL

# The access control granted to clients on the named pipe created by the MySQL Server.
# named-pipe-full-access-group=

# The TCP/IP Port the MySQL Server will listen on
port=3306
bind-address=*
```

Тут же можно можно настроить порт прослушивания сервера (выделено на скриншоте), если появилась такая необходимость уже после установки и первоначальной настройки сервера.

Обратите внимание!

На скриншоте выше выделен еще один параметр **skip-networking**, который в свою очередь может запретить любые TCP/IP подключения. Чтобы TCP/IP подключения осуществлялись этот параметр должен быть закомментирован, как на скриншоте.

Подробнее о настройке конфигурационного файла можно почитать тут: <https://dev.mysql.com/doc/refman/5.7/en/option-files.html>, а о возможных параметрах тут: <https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html>.

Обратите внимание!

Также стоит уделить внимание параметру **max_connections**, который определяет максимальное количество одновременных подключений к серверу. По умолчанию, это 151. Если количество касс, которые будут использовать данный сервер для контроля КМ, превышает значение 151, то лучше увеличить значение параметра **max_connections** в редактируемом в данный момент конфигурационном файле. Данный параметр по умолчанию должен содержаться в данном файле, конкретнее в секции **[mysqld]**.

Сохраняем все изменения и перезапускаем сервер (службу) MySQL.

PostgreSQL

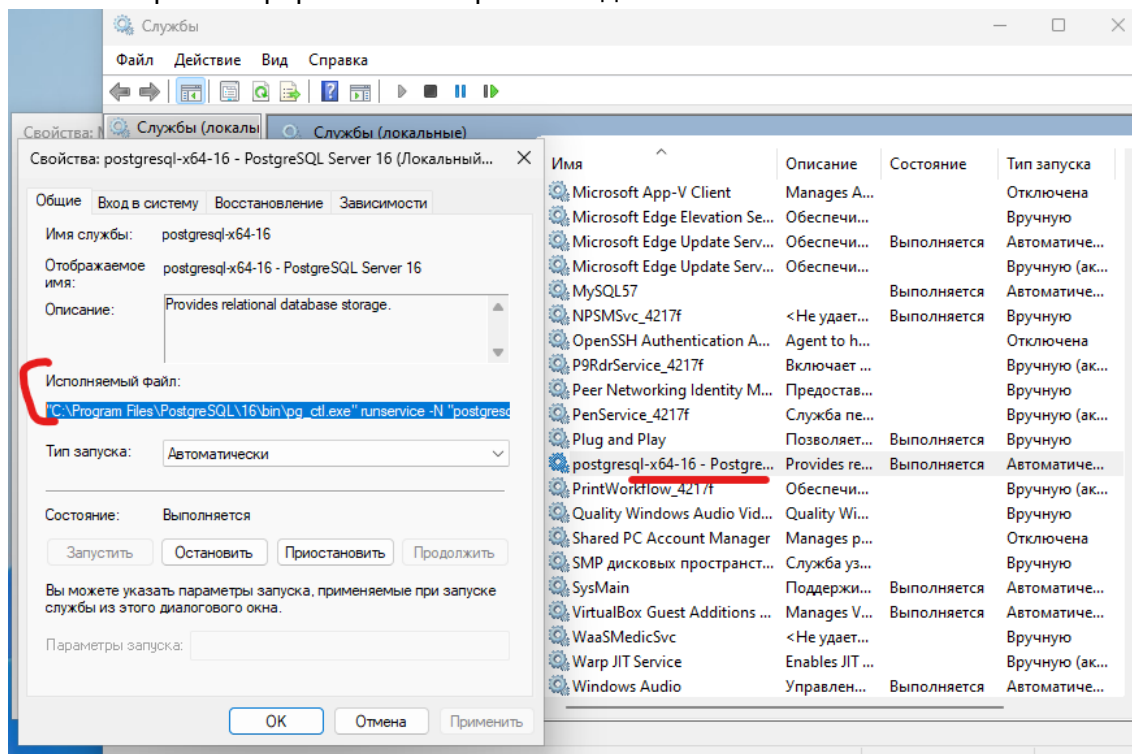
Для того, чтобы PostgreSQL прослушивал подключение с внешних рабочих машин нужно отредактировать конфигурационный файл сервера - **postgresql.conf**. Узнать о его расположении и формате можно тут <https://postgrespro.ru/docs/postgresql/12/config-setting#CONFIG-SETTING->

[CONFIGURATION-FILE](#) (лучше официальной документации объяснить не получится). Если же вам лень читать документацию, то дальше описан “короткий путь”.

Проще всего узнать расположение файла **postgresql.conf**, который использует ваш сервер - это:

1. открыть список служб Windows, найти соответствующую службу сервера PostgreSQL
2. кликнуть на найденную службу в списке правой клавишей мыши
3. в появившемся списке выбрать пункт **Свойства**

Откроется форма как на скриншоте далее:



В этой форме нас интересует свойство **Исполняемый файл**, в котором будет значиться строка подобная этой (в зависимости от версии установленного сервера и места установки выделенные зеленым цветом участки строки могут отличаться):

"C:\Program Files\PostgreSQL\16\bin\pg_ctl.exe" runservice -N "postgresql-x64-16" -D "C:\Program Files\PostgreSQL\16\data" -w

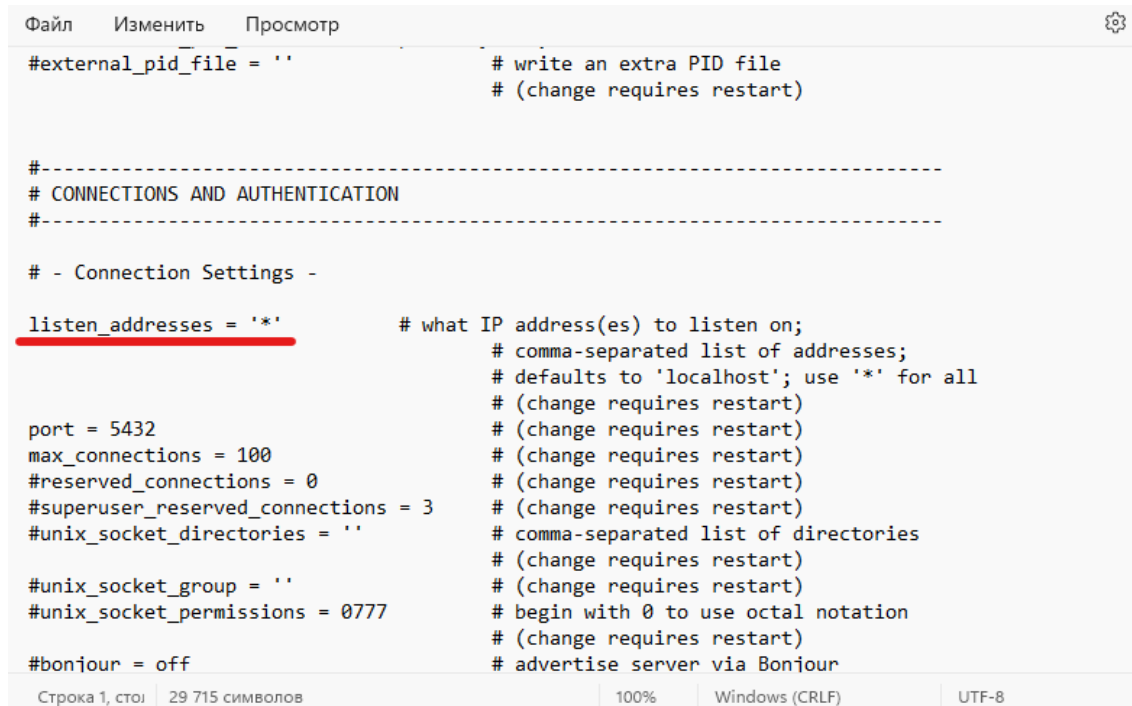
В указанной строке нас интересует выделенная кавычками часть после **-D**, в нашем случае это **"C:\Program Files\PostgreSQL\16\data"**. Это путь до директории с данными сервера, в корне которой и лежит нужный файл **postgresql.conf**.

После того, как мы установили расположение конфигурационного файла, его можно открыть в любом текстовом редакторе (не редакторе документов, вроде WORD, а текстовым - вроде блокнота).

Обратите внимание!

По формате конфигурационного файла строки начинающиеся с символа **#** или являются комментариями и игнорируются сервером при установке значений параметров.

В самом конфигурационном файле **postgresql.conf** также имеется краткое описание того, как его надо заполнять. Нас интересуют строки выделенные на скриншоте:



```
Файл  Изменить  Просмотр  [Settings Icon]

#external_pid_file = ''           # write an extra PID file
                                   # (change requires restart)

#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

listen_addresses = '*'           # what IP address(es) to listen on;
                                   # comma-separated list of addresses;
                                   # defaults to 'localhost'; use '*' for all
                                   # (change requires restart)
port = 5432                       # (change requires restart)
max_connections = 100             # (change requires restart)
#reserved_connections = 0         # (change requires restart)
#superuser_reserved_connections = 3 # (change requires restart)
#unix_socket_directories = ''     # comma-separated list of directories
                                   # (change requires restart)
#unix_socket_group = ''          # (change requires restart)
#unix_socket_permissions = 0777  # begin with 0 to use octal notation
                                   # (change requires restart)
#bonjour = off                   # advertise server via Bonjour

Строка 1, столб 29 715 символов | 100% | Windows (CRLF) | UTF-8
```

При помощи параметра **listen_addresses** можно указывать с каких ip-адресов разрешено подключение к этому серверу. По умолчанию **listen_addresses = 'localhost'**, что в свою очередь разрешает подключение к серверу ТОЛЬКО с той машины, на которой он установлен. Для разрешения подключения ДЛЯ ВСЕХ (с любого ip-адреса) нужно указать значение ***** (звездочка) в одинарных кавычках, как на скриншоте выше. Также с помощью этого параметра можно более гибко настроить доступ для внешних подключений. Подробно об этом можно почитать тут:

<https://postgrespro.ru/docs/postgrespro/9.5/runtime-config-connection#guc-listen-addresses>.

Обратите внимание!

Также стоит уделить внимание параметру **max_connections**, который определяет максимальное количество одновременных подключений к серверу. По умолчанию, это 100. Если количество касс, которые будут использовать данный сервер для контроля КМ, превышает значение 100, то лучше увеличить значение параметра **max_connections** в редактируемом в данный момент конфигурационном файле. Данный параметр по умолчанию должен содержаться в данном файле.

На скриншоте выше этот параметр имеется и находится сразу после параметра **port**, который находится сразу после **listen_addresses**.

Сохраняем изменения и перезапускаем сервер (службу) PostgreSQL.

Настройка прав пользователя для удаленного подключения

Помимо настроек для разрешения внешних подключений SQL-сервер, как правило, также позволяет настраивать возможность удаленного подключения для каждого конкретного пользователя СУБД.

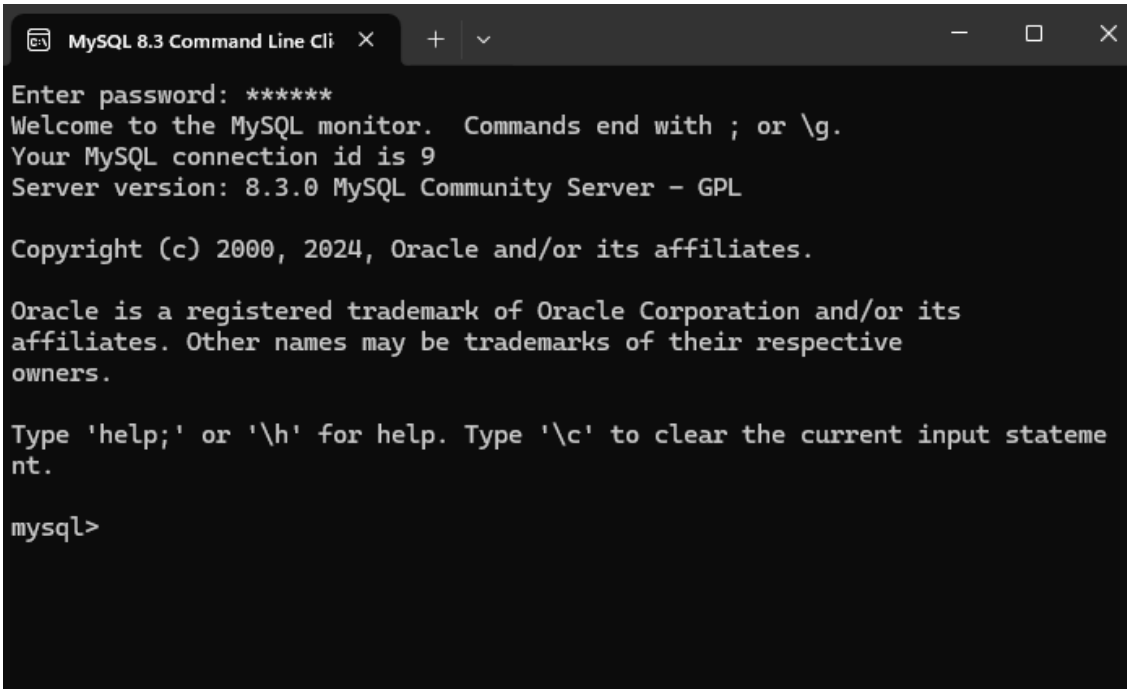
В целях безопасности, возможность удаленного подключения от имени суперпользователя (**root** в случае MySQL и **postgres** в случае PostgreSQL) запрещена.

Так как в нашем случае использование SQL-сервера осуществляется для обеспечения сохранности данных и синхронизации этих данных между несколькими кассами, зная, что никакой секретной информации в базе мы хранить не будем и с учетом того, что наш сервер будет использоваться **ТОЛЬКО** для контроля КМ по кранам (или просто контроля КМ), то требованиями безопасности можно пренебречь.

MySQL

Для того, чтобы можно было удаленно подключиться к серверу MySQL пользователю **root**, надо изменить параметры в таблице **mysql.user**.

Запускаем утилиту **MySQL <версия сервера> Command Line Client** (где <версия сервера> - это версия установленного сервера, например для 8.3 утилита имеет наименование **MySQL 8.3 Command Line Client**), вводим пароль пользователя **root** и нажимаем **Enter**. Итог на скриншоте ниже.



```
MySQL 8.3 Command Line Cli
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input stateme
nt.

mysql>
```

Далее, чтобы узнать текущий доступ пользователя root (можно ли ему подключаться удаленно) введем следующую команду:

```
SELECT user, host FROM mysql.user WHERE user='root';
```

Итог выполнения данной команды отражен в скриншоте ниже:

```
MySQL 8.3 Command Line Cli X + v - □ X
Your MySQL connection id is 10
Server version: 8.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input stateme
nt.

mysql> SELECT user, host FROM mysql.user WHERE user='root';
+-----+-----+
| user | host      |
+-----+-----+
| root | localhost |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Видим таблицу (user, host) с нашим пользователем **root**, для которого в поле **host** указано **localhost**, а это означает, что данному пользователю можно подключиться к серверу **MySQL** ТОЛЬКО локально (с той машины, на которой установлен сервер).

Далее нам надо изменить значение поля **host** с **localhost** на символ **%** для нашего пользователя **root**. Для этого надо выполнить следующую команду:

UPDATE mysql.user SET host='%' WHERE user='root';

Если команда выполнялась, то вы увидите следующий вывод утилиты:

```
MySQL 8.3 Command Line Cli X + v - □ X

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input stateme
nt.

mysql> SELECT user, host FROM mysql.user WHERE user='root';
+-----+-----+
| user | host      |
+-----+-----+
| root | localhost |
+-----+-----+
1 row in set (0.00 sec)

mysql> UPDATE mysql.user SET host='%' WHERE user='root';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
```

Для продолжения работы, в связи с изменением значения **host** для пользователя **root**, под которым мы сейчас подключились к серверу, используя утилиту, необходимо осуществить переподключение:

1. закройте текущий экземпляр утилиты
2. заново запустите эту же утилиту

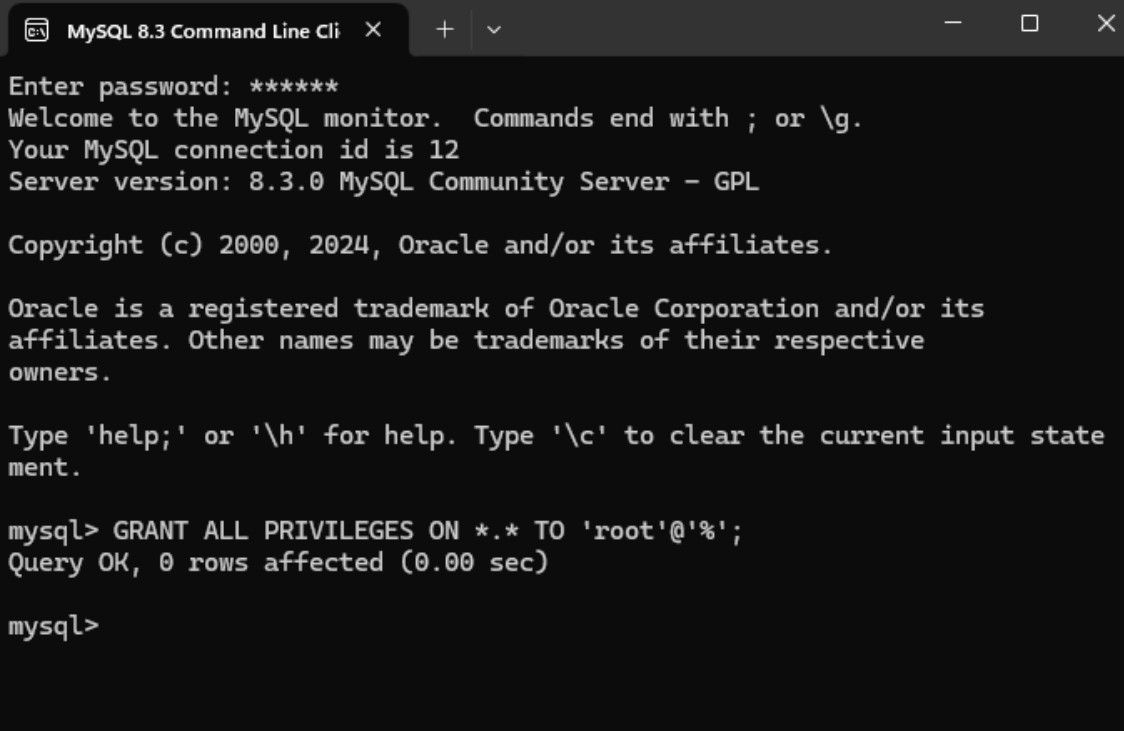
Далее, на всякий случай, мы заново выдадим пользователю root все привилегии (подробнее о привилегиях можно почитать тут - <https://www.opennet.ru/docs/RUS/mysqladm/userman.html>). Для выдачи всех привилегий надо ввести следующую команду:

GRANT ALL PRIVILEGES ON *.* TO 'root'@'<host>';

где **<host>** - это текущее значение поля **host** для нашего пользователя **root**, которое после выполнения предыдущей команды имеет значение **%**. В итоге команда, которую нам надо выполнить должна выглядеть вот так:

GRANT ALL PRIVILEGES ON *.* TO 'root'@'%';

После выполнения данной команды должен быть такой же вывод, как на скриншоте далее:



```
MySQL 8.3 Command Line Cli x + v - □ x
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input state
ment.

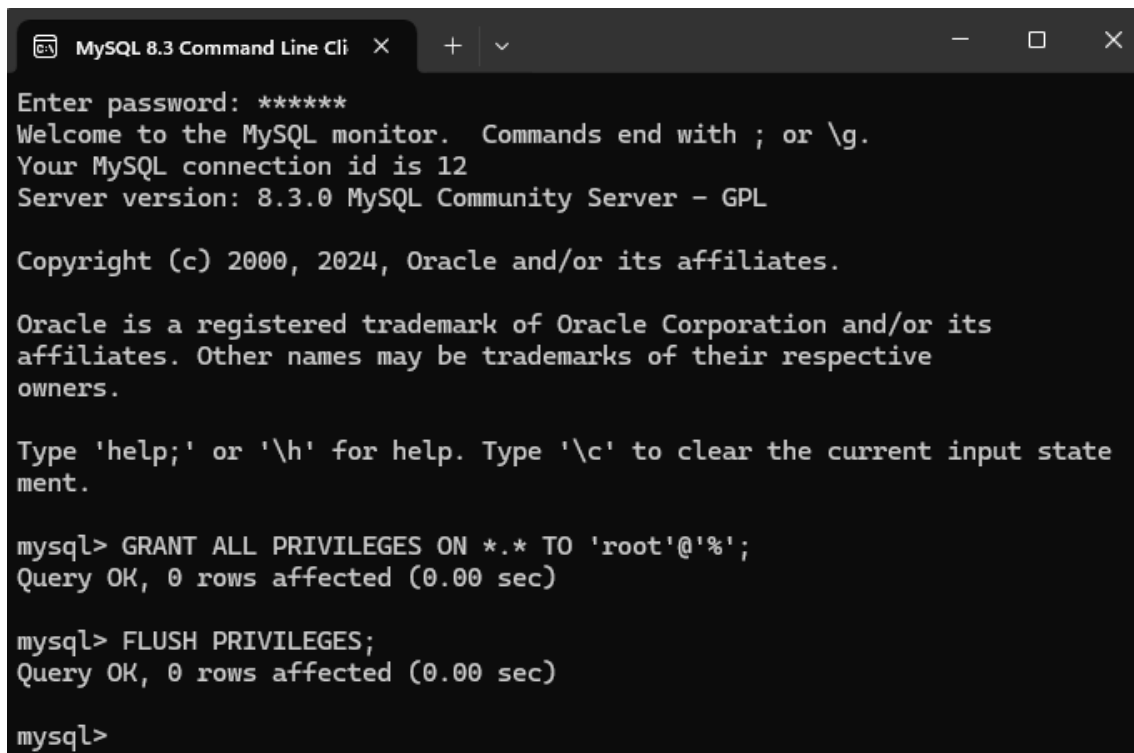
mysql> GRANT ALL PRIVILEGES ON *.* TO 'root'@'%';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

Далее нам надо сообщить серверу о том, что нужно перезагрузить таблицы предоставления привилегий (считай, сбросить кэш привилегий). Для этого необходимо выполнить следующую команду:

FLUSH PRIVILEGES;

После выполнения данной команды должен быть такой же вывод, как на скриншоте ниже:



```
MySQL 8.3 Command Line Cli X + v - □ X
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\\h' for help. Type '\\c' to clear the current input state
ment.

mysql> GRANT ALL PRIVILEGES ON *.* TO 'root'@'%';
Query OK, 0 rows affected (0.00 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

Перезапускаем сервер (службу) MySQL, чтобы быть уверенными в том, что изменения применились.

PostgreSQL

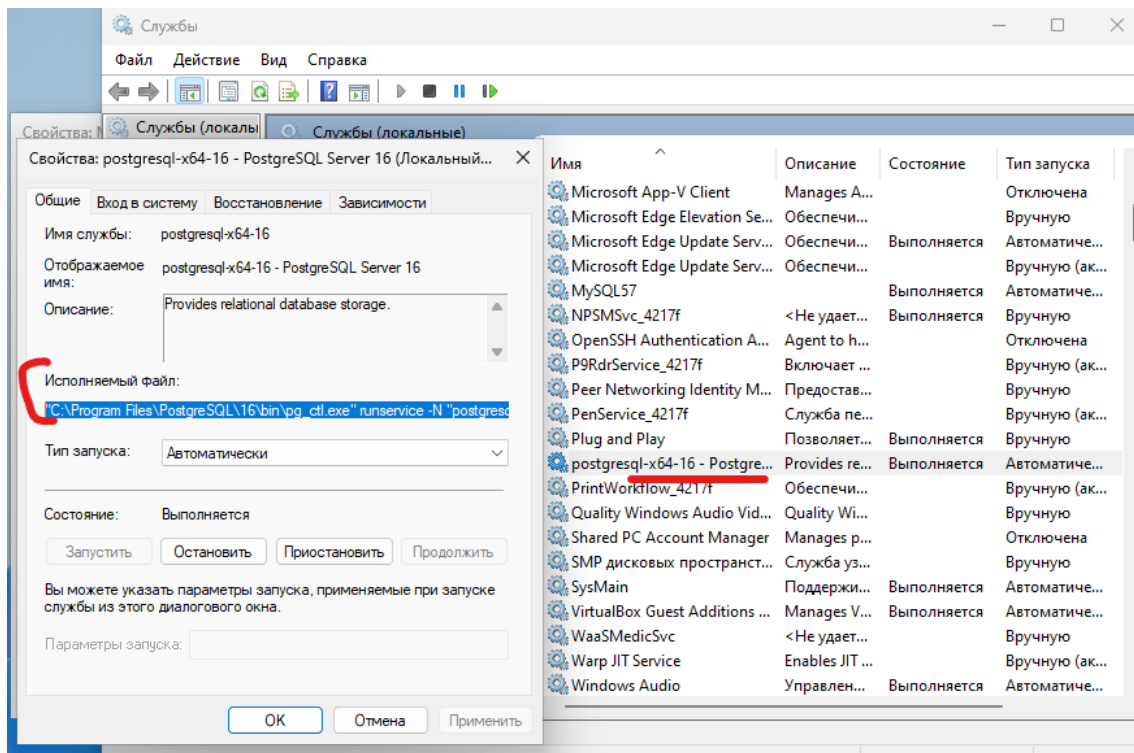
Для того, чтобы к серверу PostgreSQL можно было подключиться удаленно пользователю **postgres** (всем пользователям) нужно отредактировать конфигурационный файл **pg_hba.conf**.

Узнать о его расположении и заполнении можно тут <https://postgrespro.ru/docs/postgrespro/10/auth-pg-hba-conf> (лучше официальной документации не получится объяснить). Если же вам лень читать документацию, то дальше описан “короткий путь”.

Проще всего узнать расположение файла **pg_hba.conf**, который использует ваш сервер - это:

1. открыть список служб Windows, найти соответствующую службу сервера PostgreSQL
2. кликнуть на найденную службу в списке правой клавишей мыши
3. в появившемся списке выбрать пункт **Свойства**

Откроется форма как на скриншоте далее:



В этой форме нас интересует свойство **Исполняемый файл**, в котором будет значиться строка подобная этой (в зависимости от версии установленного сервера и места установки выделенные зеленым участки строки могут отличаться):

"C:\Program Files\PostgreSQL\16\bin\pg_ctl.exe" runservice -N "postgresql-x64-16" -D "C:\Program Files\PostgreSQL\16\data" -w

В указанной строке нас интересует выделенная кавычками часть после **-D**, в нашем случае это **"C:\Program Files\PostgreSQL\16\data"**. Это путь до директории с данными сервера, в корне которой и лежит нужный нам файл **pg_hba.conf**.

После того, как мы установили расположение конфигурационного файла, его можно открыть в любом текстовом редакторе (не редакторе документов, вроде WORD, а текстовым - вроде блокнота).

Обратите внимание!

По формате конфигурационного файла строки начинающиеся с символа **#** или являются комментариями и игнорируются сервером при установке значений параметров.

В самом конфигурационном файле **pg_hba.conf** также имеется краткое описание того, как его надо заполнять. Нас интересуют строки выделенные на

скриншоте:

```
# -----  
#  
# If you want to allow non-local connections, you need to add more  
# "host" records. In that case you will also need to make PostgreSQL  
# listen on a non-local interface via the listen_addresses  
# configuration parameter, or via the -i or -h command line switches.  
  
# TYPE DATABASE USER ADDRESS METHOD  
  
# "local" is for Unix domain socket connections only  
local all all scram-sha-256  
# IPv4 local connections:  
host all all 127.0.0.1/32 scram-sha-256  
# IPv6 local connections:  
host all all ::1/128 scram-sha-256  
# Allow replication connections from localhost, by a user with the  
# replication privilege.  
local replication all scram-sha-256  
host replication all 127.0.0.1/32 scram-sha-256  
host replication all ::1/128 scram-sha-256
```

Первая выделенная строка отвечает за настройку подключений по IPv4, а вторая за IPv6. На данном скриншоте настроено подключение по IPv4 и IPv6 ТОЛЬКО для локальной машины (127.0.0.1 = localhost, а ::1 - это IPv6-аналог "127.0.0.1", то есть тоже localhost).

Чтобы разрешить IPv4-подключения ВСЕМ пользователям ко ВСЕМ базам данных, нужно изменить в строке "**host all all 127.0.0.1/32 ...**" заменить часть **127.0.0.1/32** на **0.0.0.0/0**, для IPv6-подключений надо заменить в строке "**host all all ::1/128**" часть **::1/128** на **::0/0**. Все это описано в официальной документации.

В итоге у вас должно получиться так:

```
# Put your actual configuration here  
# -----  
#  
# If you want to allow non-local connections, you need to add more  
# "host" records. In that case you will also need to make PostgreSQL  
# listen on a non-local interface via the listen_addresses  
# configuration parameter, or via the -i or -h command line switches.  
  
# TYPE DATABASE USER ADDRESS METHOD  
  
# "local" is for Unix domain socket connections only  
local all all scram-sha-256  
# IPv4 local connections:  
host all all 0.0.0.0/0 scram-sha-256  
# IPv6 local connections:  
host all all ::0/0 scram-sha-256  
# Allow replication connections from localhost, by a user with the  
# replication privilege.  
local replication all scram-sha-256  
host replication all 127.0.0.1/32 scram-sha-256  
host replication all ::1/128 scram-sha-256
```

Сохраняем файл и перезапускаем сервер (службу) PostgreSQL.

Настройки компьютера с ПО “Кассир 5”

Установка ODBC-драйвера

Установка ODBC-драйвера MySQL

Для установки ODBC-драйвера надо скачать соответствующий установщик по ссылке: <https://dev.mysql.com/downloads/connector/odbc/> или <https://downloads.mysql.com/archives/c-odbc/> (по этой ссылке можно получить архивные версии драйвера).

Обратите внимание!

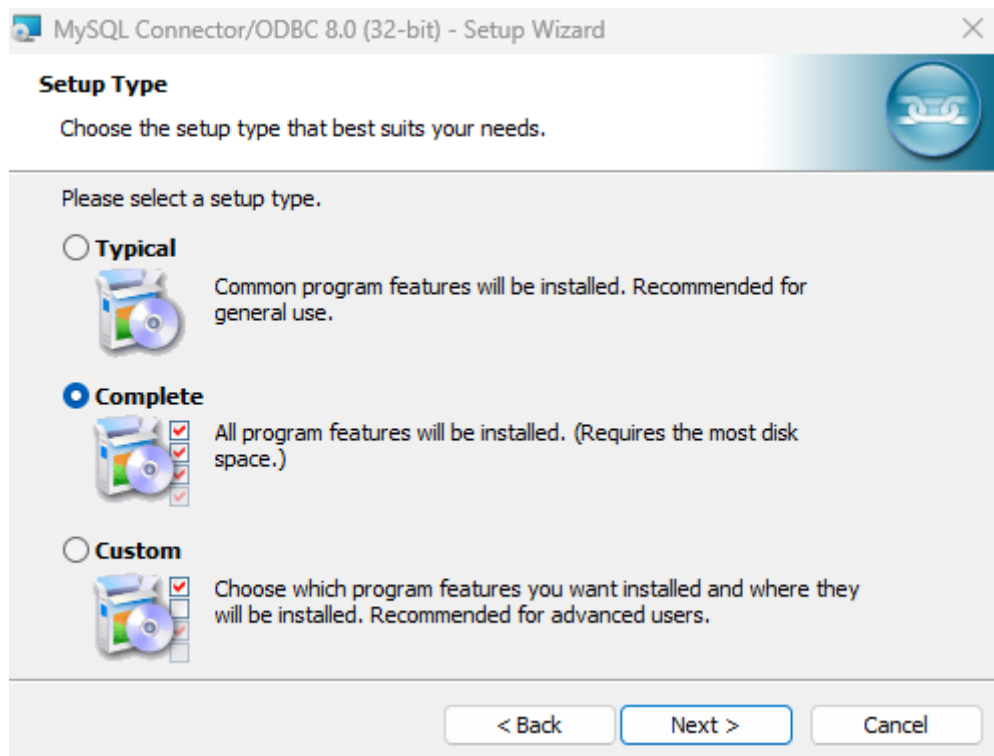
При выборе установщика драйвера необходимо учитывать, что для работы с ПО “Кассир 5” необходима 32-битная версия драйвера. Обратите внимание, что не все версии драйвера имеют 32-битную версию (по первой ссылке максимальная версия драйвера с поддержкой 32-битной архитектуры - **8.0.36**).

Если при выборе версии **8.0.36** в списке нет доступных для скачивания 32-битных вариантов установщиков (смотри скриншот далее), то выберите версию **8.3.0** и затем обратно верните **8.0.36** - это баг сайта.

Operating System	Version	Size	Action
Windows (x86, 32-bit), MSI Installer	8.0.36	11.5M	Download
(mysql-connector-odbc-8.0.36-win32.msi) MD5: 4228a05f487fb59ca9108f41da200ceb Signature			
Windows (x86, 32-bit), ZIP Archive	8.0.36	11.3M	Download
(mysql-connector-odbc-noinstall-8.0.36-win32.zip) MD5: 8adb42af93de7e6b1464d8fc1713a1d9 Signature			
Windows (x86, 32-bit), ZIP Archive Debug Binaries & Test Suite	8.0.36	42.8M	Download
(mysql-connector-odbc-noinstall-8.0.36-win32-debug.zip) MD5: 338ded7cdfe06813c0e80d7746b2bda9 Signature			

We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.

После того, как архив установщик скачан, можно его запустить.
При установке драйвера нет каких-либо особенностей.



На этом этапе выбираем пункт **Complete** и нажимаем **Next >** установка не будет завершена.

Установка ODBC-драйвера PostgreSQL (psqlodbc)

Для установки ODBC-драйвера надо скачать соответствующий установщик по ссылке: <https://www.postgresql.org/ftp/odbc/versions/msi/>

Обратите внимание!

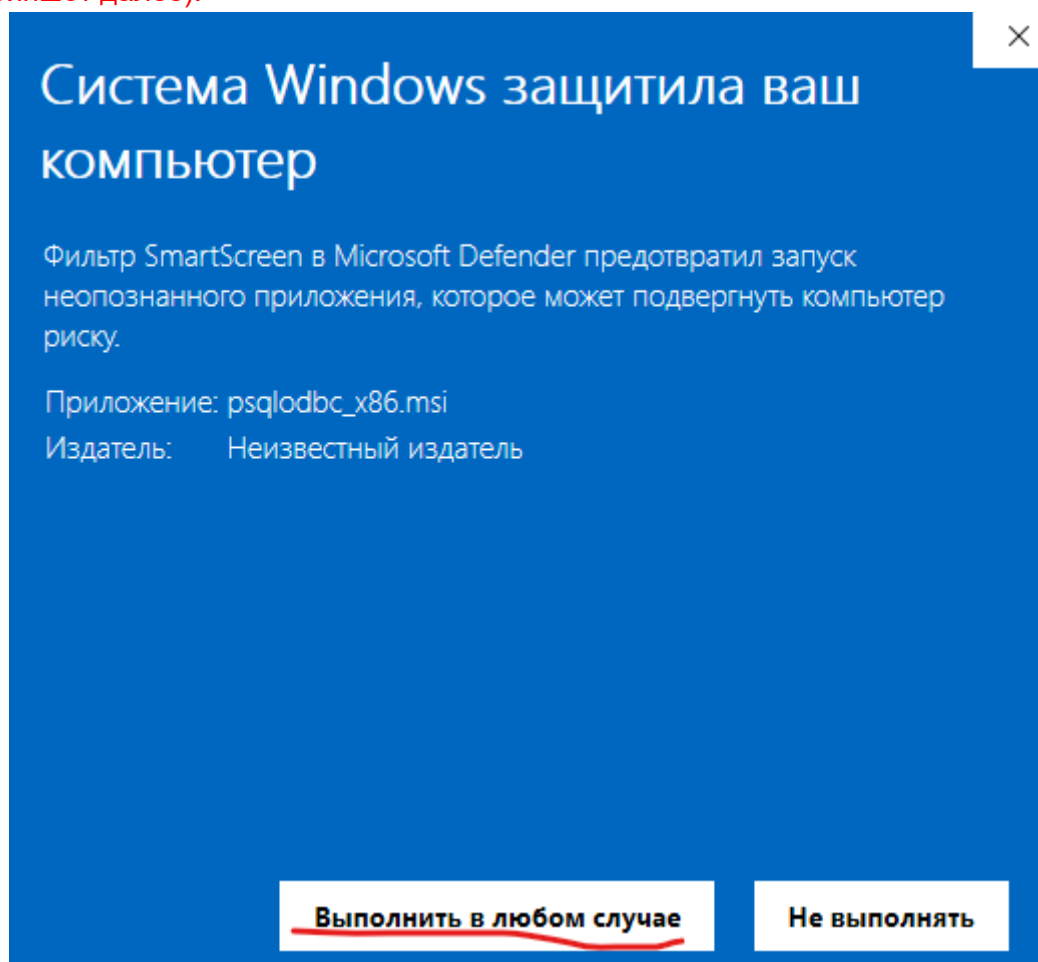
При выборе установщика драйвера необходимо учитывать, что для работы с ПО “Кассир 5” необходима 32-битная версия драйвера. Поэтому из представленного по ссылке списка zip-архивов выбираем тот, в наименовании которого содержится суффикс **x86**.

Например, при написании данной инструкции использовался архив **psqlodbc_16_00_0000-x86.zip**.

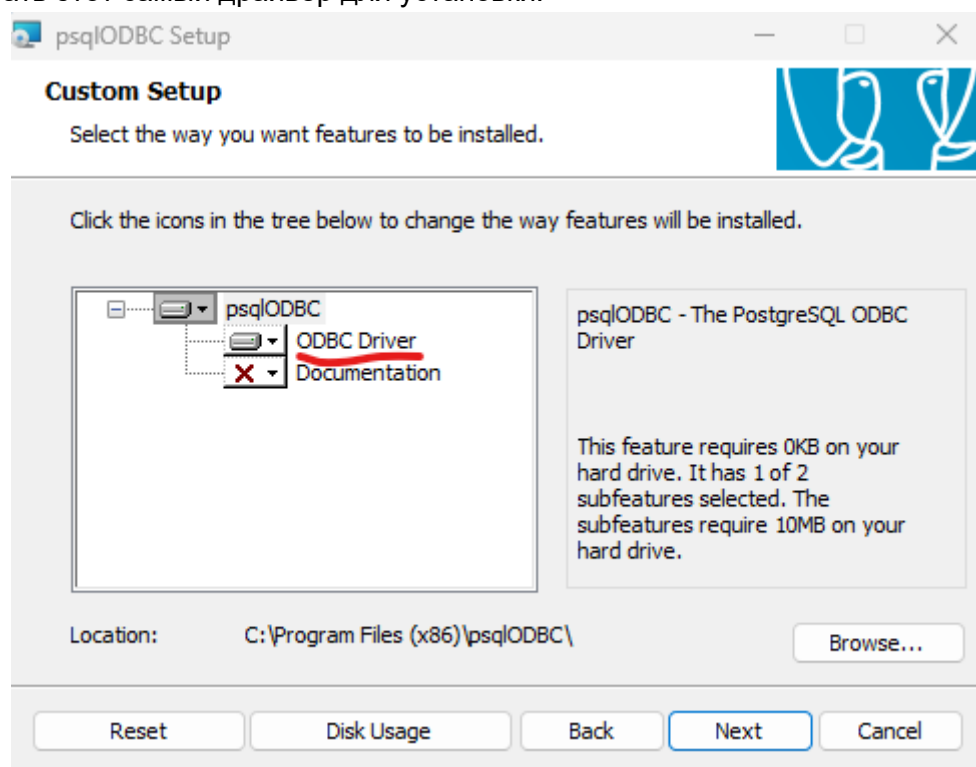
После того, как архив с установщиком скачан, можно его распаковать или сразу из него запустить файл установщика **psqlodbc_x86.msi**.

Обратите внимание!

При попытке запустить установщик Windows может остановить выполнение “неопознанного приложения”. Ничего страшного, можно просто проигнорировать это “предупреждение” и “Выполнить в любом случае” (см. скриншот далее).



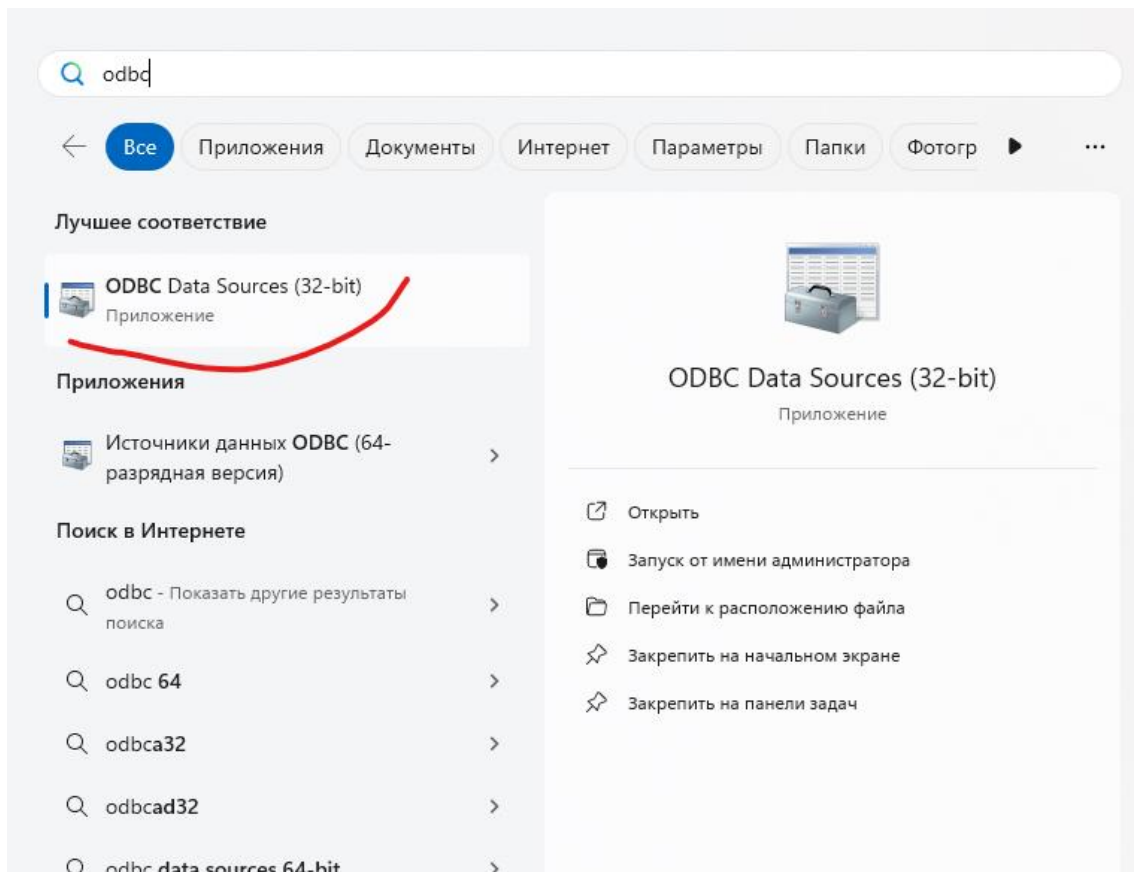
При установке драйвера нет каких-либо особенностей, главное, чтобы на этапе выбора компонентов для установки (см. скриншот далее) не забыть выбрать этот самый драйвер для установки:



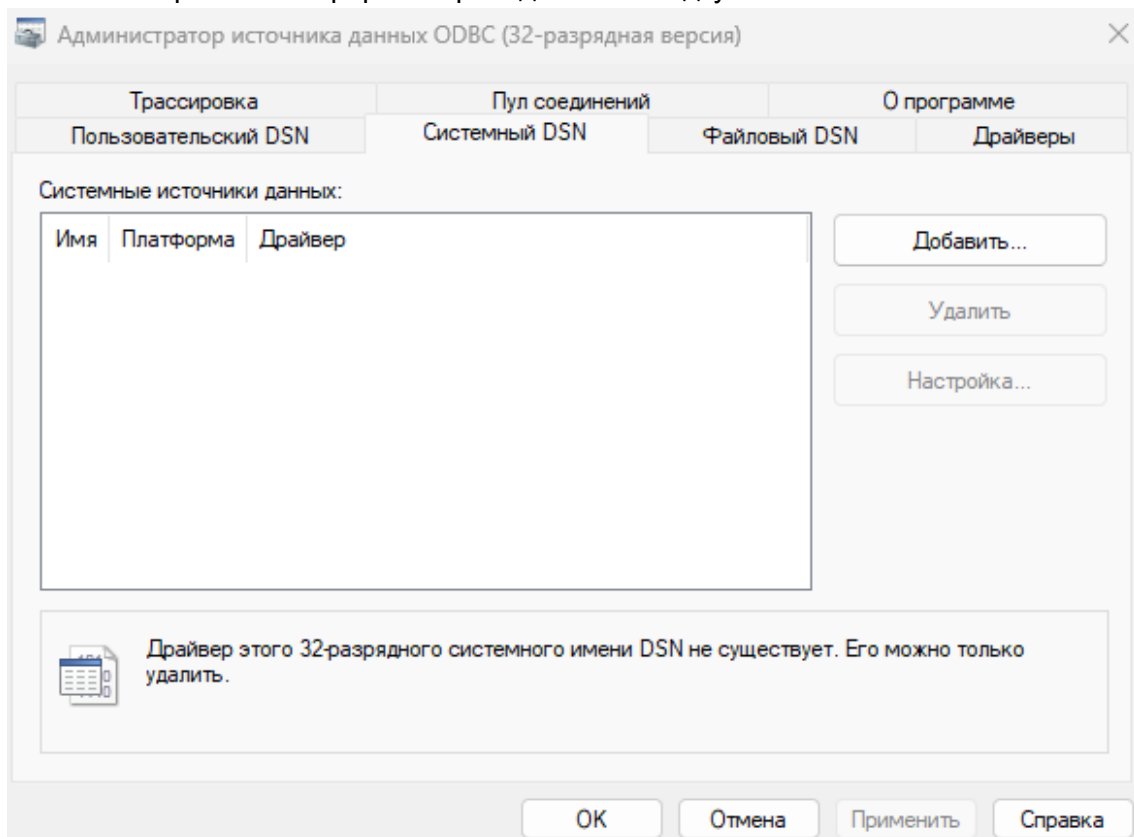
Настройка источников данных ODBC

Для настройки источника данных надо:

1. Нажимаем кнопку пуск
2. Вводим строку **ODBC**
3. Выбираем приложение **ODBC Data Sources (32-bit)**, как на скрине далее

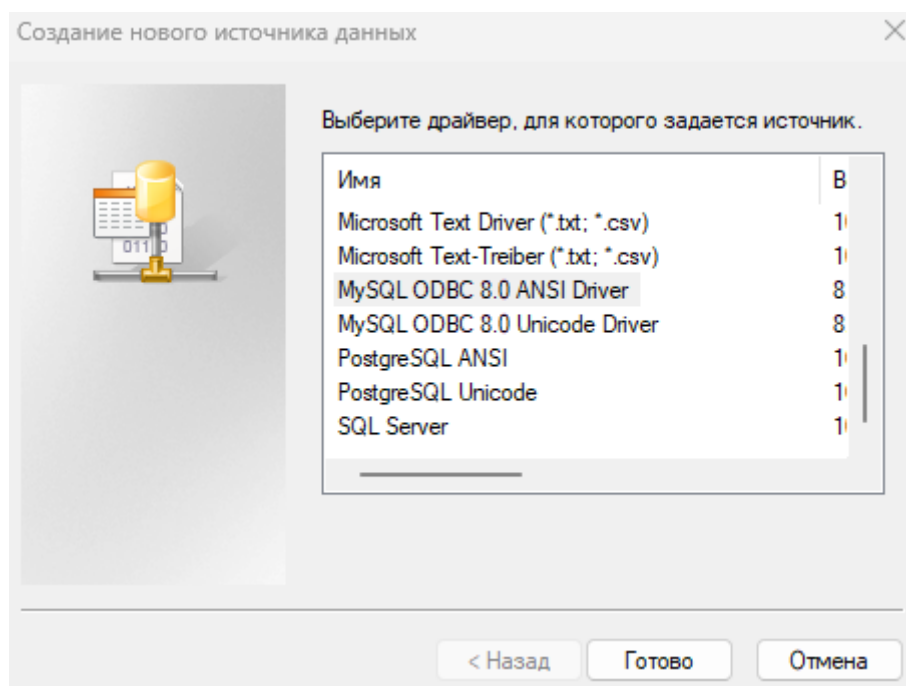


В открывшейся форме переходим во вкладку **Системные DSN**:

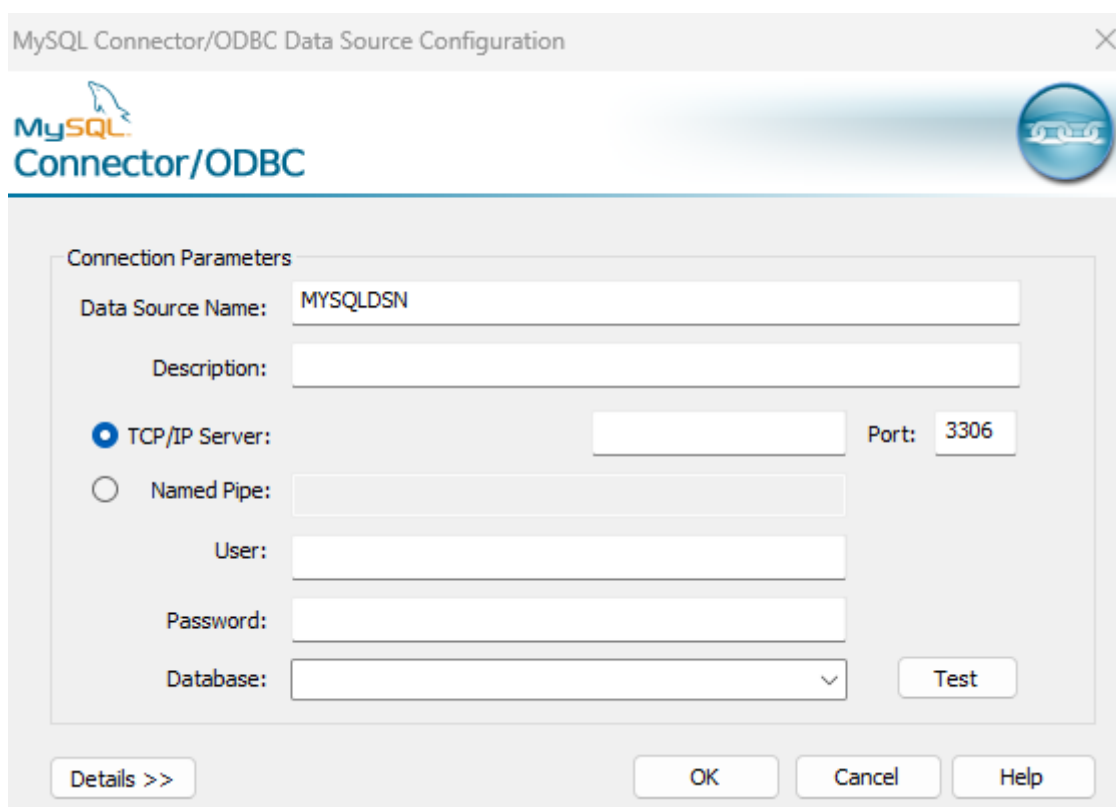


Далее нажимаем **Добавить...**

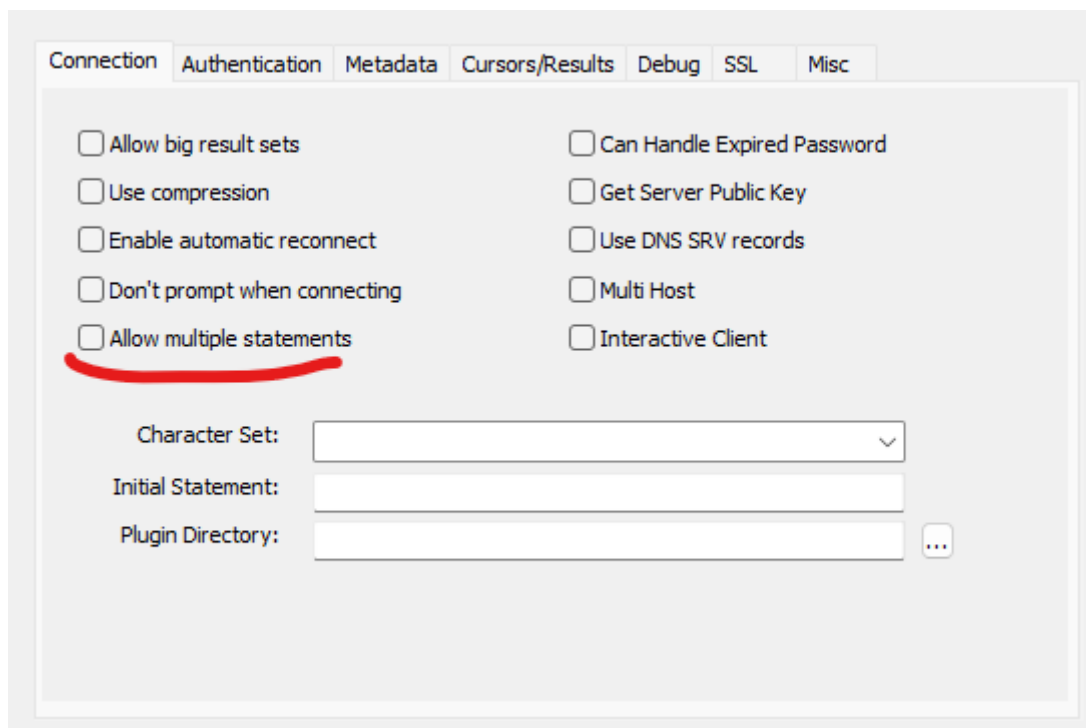
Настройка источника данных MySQL



Выбираем из списка драйверов строку с наименованием **MySQL ODBC <версия драйвера> ANSI Driver** и нажимаем **Готово**.



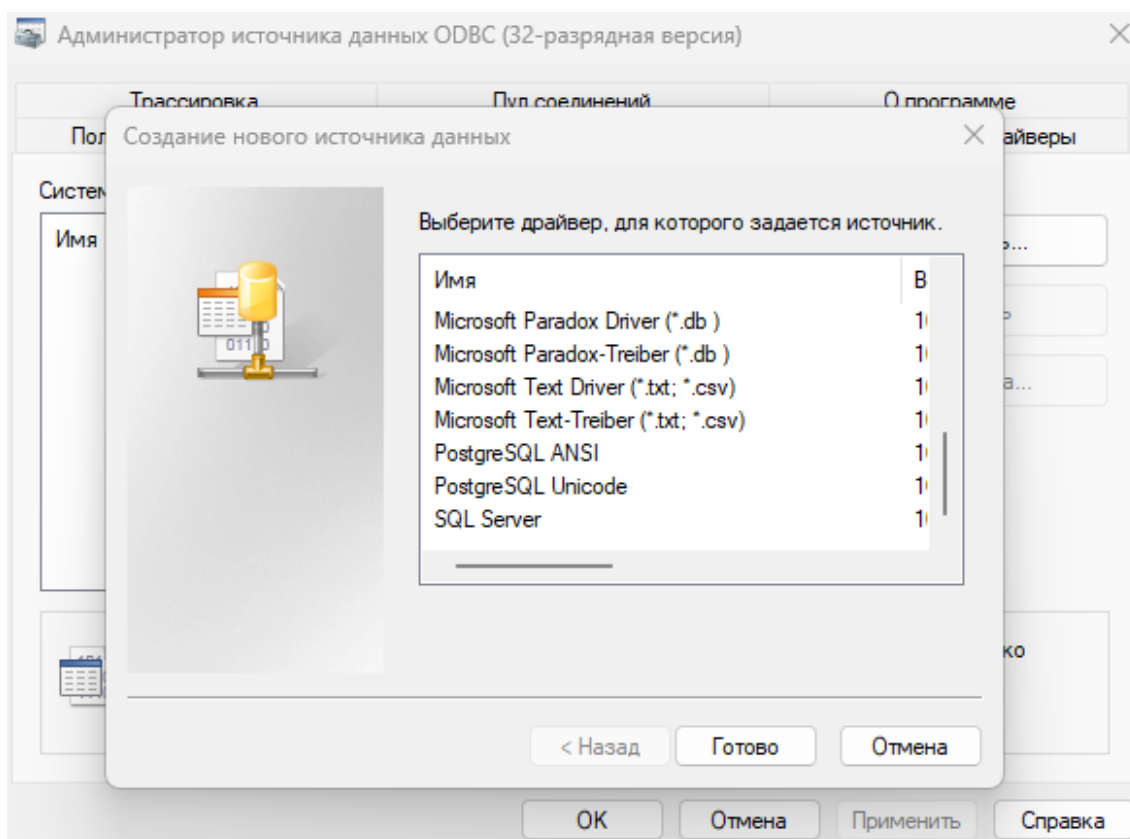
Тут в поле **Data Source Name** введите следующую строку **MYSQLDSN**, а в поле **Port** - номер порта, который указывали при установке СУБД MySQL, по умолчанию **3306**. Далее нажимаем кнопку **Details >>**.



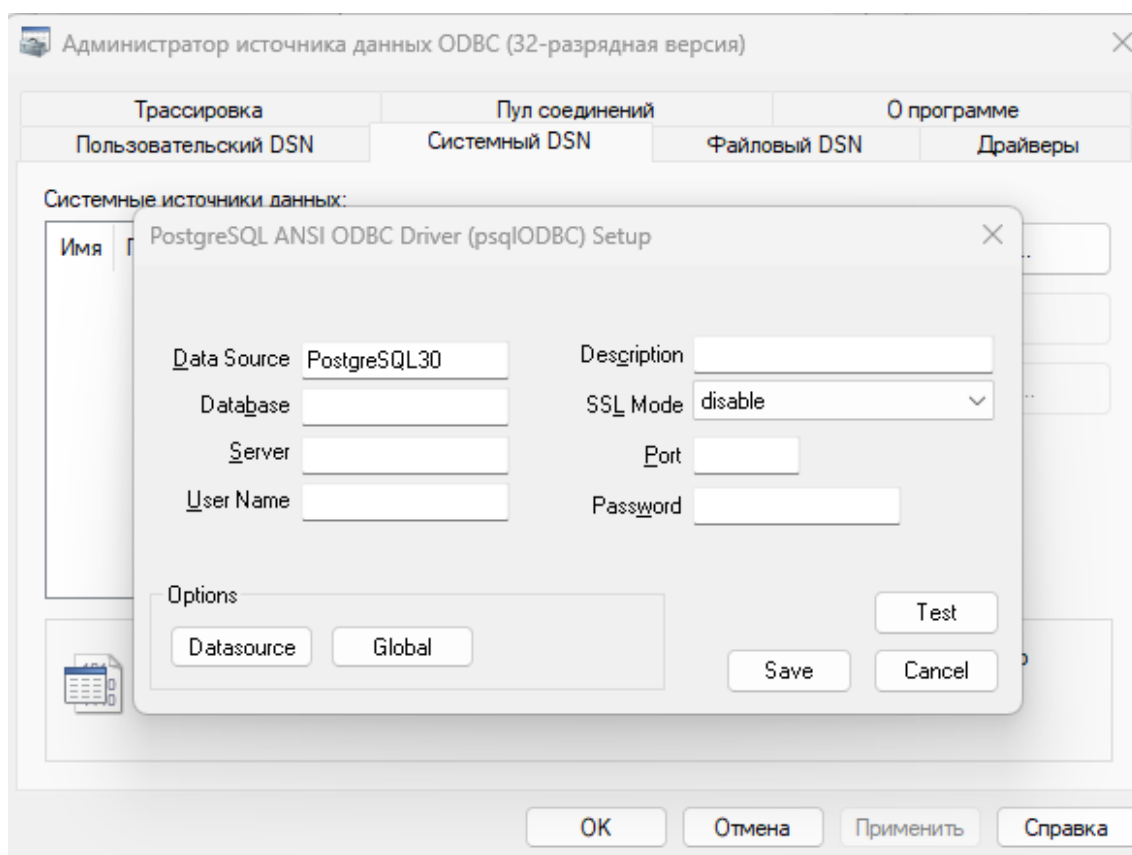
В развернувшейся панели вкладок переходим к вкладке **Connection** и ставим флажок **Allow multiple statements**.

Остальные поля можно не заполнять, так как эти данные указываются в настройках **Кассир 5**. Нажимаем **Save**. Готово! Можно переходить к настройкам **Кассир 5**.

Настройка источника данных PostgreSQL



Выбираем из списка драйверов строку с наименованием **PostgreSQL ANSI** и нажимаем **Готово**.



Тут в поле **Data source** введите следующую строку **PostgreSQL30**, а в поле **Port** - номер порта, который указывали при установке СУБД PostgreSQL, по умолчанию **5432**. Остальные поля можно не заполнять, так как эти данные указываются в настройках **Кассир 5**. Нажимаем **Save**. Готово! Можно переходить к настройкам **Кассир 5**.

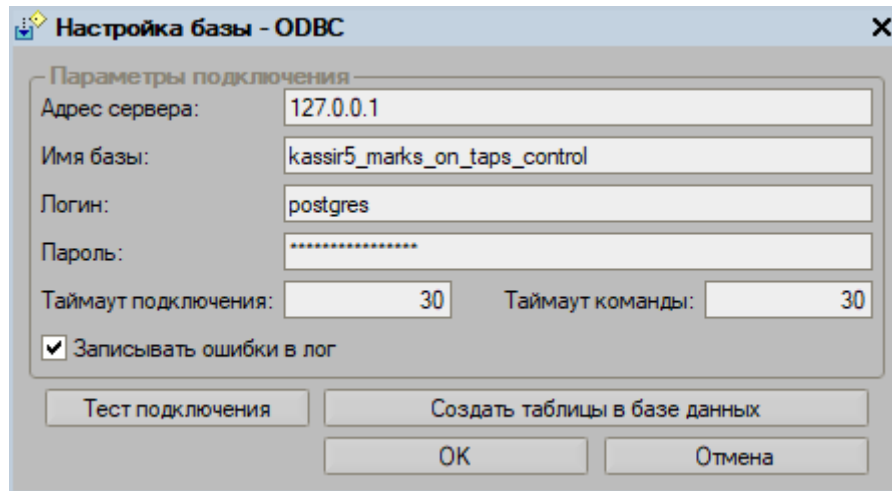
Обратите внимание!

Поле **Port** обязательно должно быть заполнено. Если при установке СУБД порт оставили по умолчанию, то в данное поле надо ввести номер порта по умолчанию - **5432**.

Настройка ПО Кассир

1. Переходим в Настройки > Режимы > Маркировка > Пиво.
2. Из списка выбора для поля **Контроль КМ по кранам** выбираем:
 - a. пункт **Внешняя БД (PostgreSQL)** для работы с PostgreSQL
 - b. пункт **Внешняя БД (MySQL)** для работы с MySQL
3. Нажимаем кнопку **Настроить** напротив поля **Контроль КМ по кранам**
4. В открывшейся форме указываем необходимые для подключения параметры:
 - a. **Адрес сервера**
 - b. **Имя базы** - имя базы указанное при создании базы (kassir5_marks_on_taps_control)
 - c. **Логин** - Имя пользователя для подключения к СУБД. Если дополнительных пользователей для Контроля КМ не создавалось, то вводим логин по умолчанию:
 - i. для PostgreSQL - **postgres**

- ii. для MySQL - root
- d. **Пароль пользователя**
- e. остальное на ваше усмотрение



- 5. Если используется СУБД MySQL, то следует нажать кнопку “Создать таблицы в базе данных”.
- 6. Если используется СУБД PostgreSQL и релиз Кассира 5.1.7.19, то нельзя нажимать кнопку “Создать таблицы в базе данных”, т.к. их нужно создавать в **SQL Shell (psql)** или при помощи **pgAdmin**.