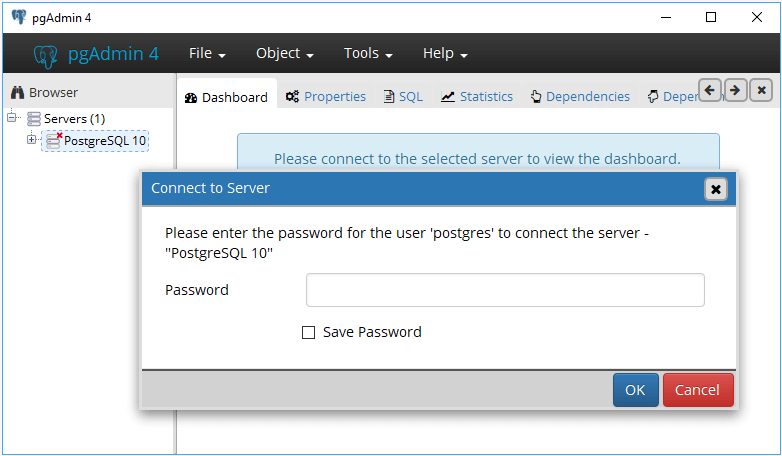
Лабораторная работа №15

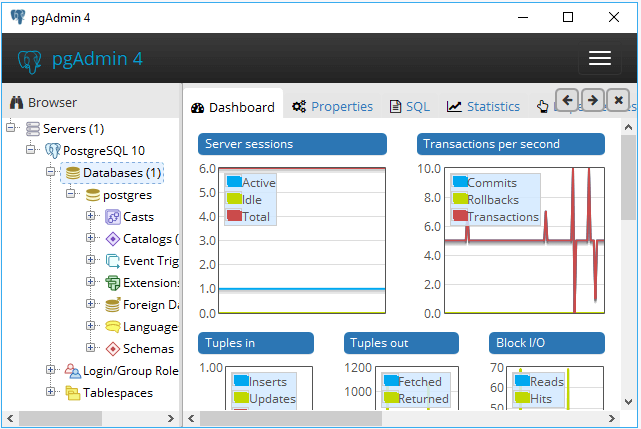
Создание и администрирование базы данных в PostgreSQL

Для создание базы данных через PgAdmin в левой части окна программы раскроем пункт Servers, который содержит набор серверов PostgreSQL. При установке последней версии устанавливается сервер, который по умолчанию имеет название PostgreSQL 10. Нажмем на этот пункт, и нам отобразится окно для ввода пароля:



Здесь необходимо ввести пароль для суперпользователя postgres, который был задан при установке PostgreSQL.

После успешного логина нам откроется содержимое сервера:

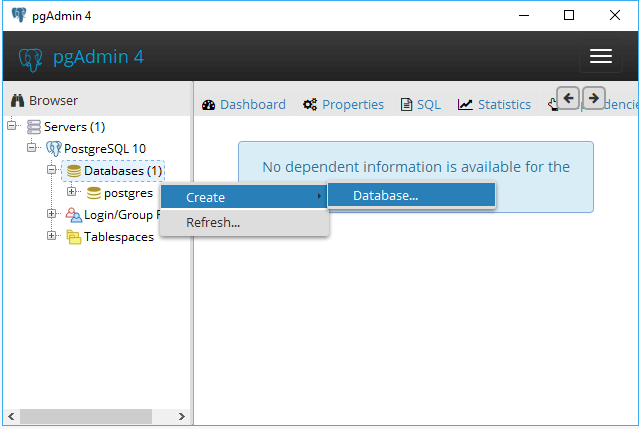


В частности, в узле Databases мы можем увидеть все имеющиеся базы данных. По умолчанию здесь есть только одна база данных - postgres.

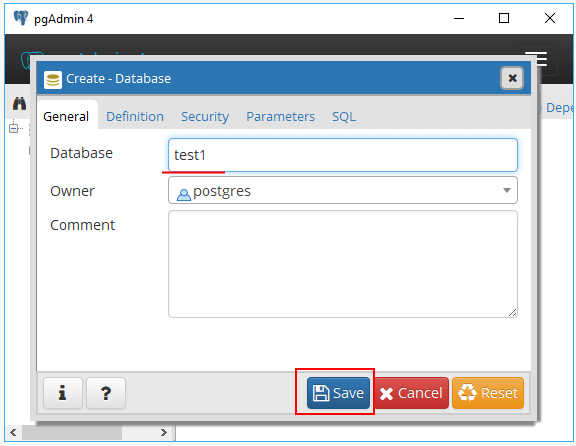
Также в правой части мы можем увидеть узел Login/Group Roles, который предназначен для управления пользователями и их ролями.

И третий узел - Tablespaces позволяет управлять местом хранения файлов баз данных.

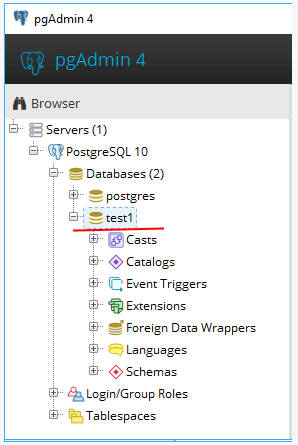
Теперь создадим свою базу данных. Для этого нажмем правой кнопкой мыши на узел Databases. И далее в контекстном меню выберем Create->Database...



После этого нам отобразится окно для создания базы данных. Введем название для БД, например, test1 и нажмем на кнопку "Save":



После этого в древовидном меню слева отобразится содержимое созданной базы данных test1:



Администрирование в PostgreSQL через pgAdmin

Создание таблиц:

Структура таблиц будет следующая:

goods – таблица будет содержать информацию о товарах:

product\_id – идентификатор товара, данное значение должно автоматически генерироваться. Столбец не может содержать значения NULL и является первичным ключом;

product\_name – наименование товара, столбец не может содержать значения NULL;

category – ссылка на категорию товара, столбец не может содержать значения NULL, но имеет значение по умолчанию, например, для случаев, когда товар еще не распределили в необходимую категорию, в этом случае товару будет присвоена категория по умолчанию («Не определена» или «Не указана»);

price – цена товара, столбец может содержать значения NULL, например, с ценой еще не определились.

categories — таблица будет содержать описание категорий товаров:

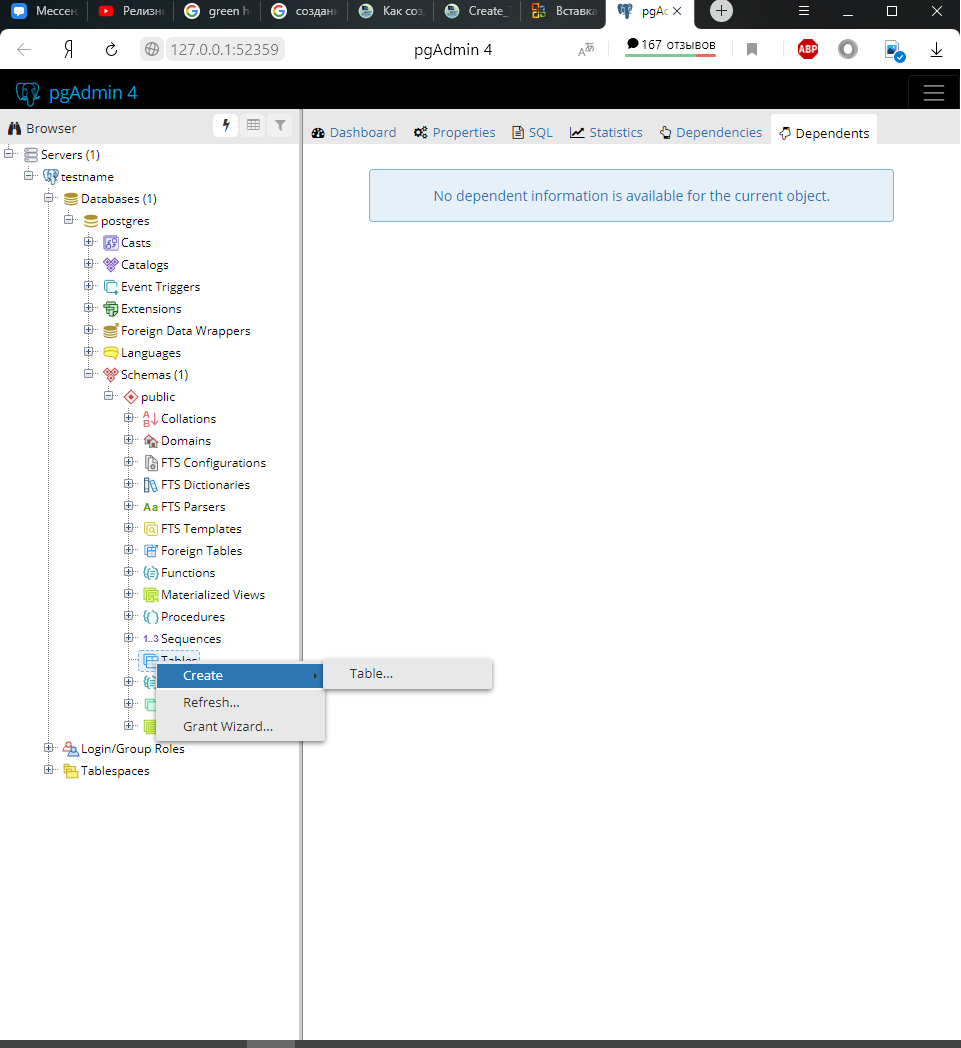
category\_id – идентификатор категории, данное значение должно автоматически генерироваться. Столбец не может содержать значения NULL и является первичным ключом;

category\_name – наименование категории, столбец не может содержать значения NULL.

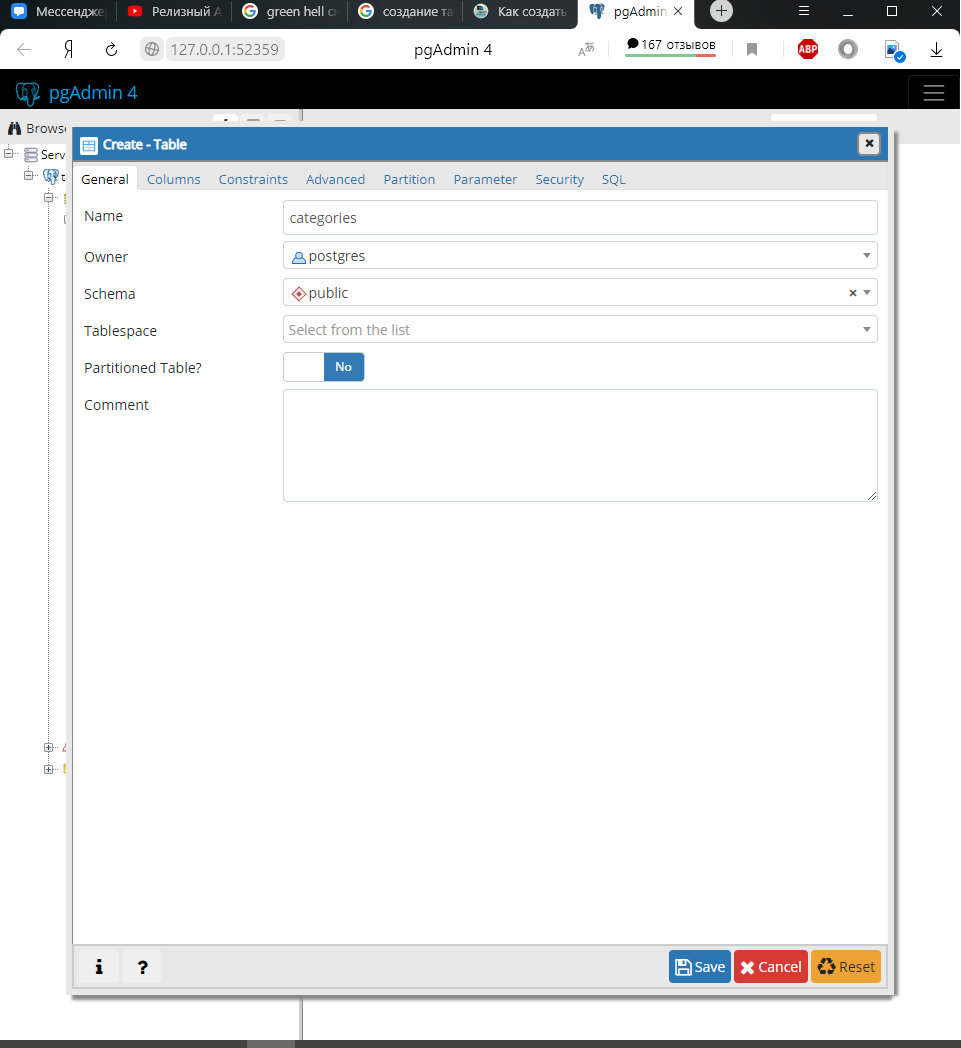
При этом внести товар с несуществующей категорией нельзя, поэтому мы добавим еще и ограничение внешнего ключа.

**Создание таблицы с помощью графического конструктора pgAdmin 4**

Чтобы создать таблицу в PostgreSQL с помощью графического конструктора pgAdmin 4, необходимо в обозревателе щёлкнуть правой кнопкой мыши по контейнеру *«Таблицы»* и выбрать пункт *«Создать -> Таблицу»*.



Затем запустится графический конструктор таблиц, где первым делом нам необходимо ввести название таблицы, сначала давайте создадим таблицу с категориями, чтобы потом в процессе создания таблицы с товарами у нас была возможность сразу определить ограничение внешнего ключа. Поэтому вводим categories.



После того как название таблицы задано, мы можем переходить к определению столбцов. Для этого необходимо перейти на вкладку *«Столбцы»* и, используя кнопку плюс «+», добавить нужные столбцы.

Сначала создаем идентификатор категории, по условиям нашей задачи он должен отвечать определенным требованиям, поэтому мы должны задать следующие свойства у столбца:

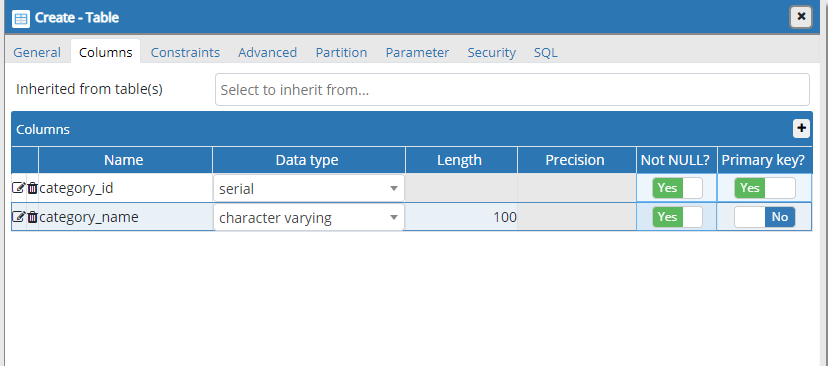
* Включить параметр «Не NULL», чтобы параметр не мог хранить значения NULL;
* Включить параметр «Первичный ключ», чтобы столбец выполнял роль первичного ключа;
* Включить идентификацию, чтобы в столбце автоматически генерировались значения.

В качестве типа данных выберем целочисленный тип integer.

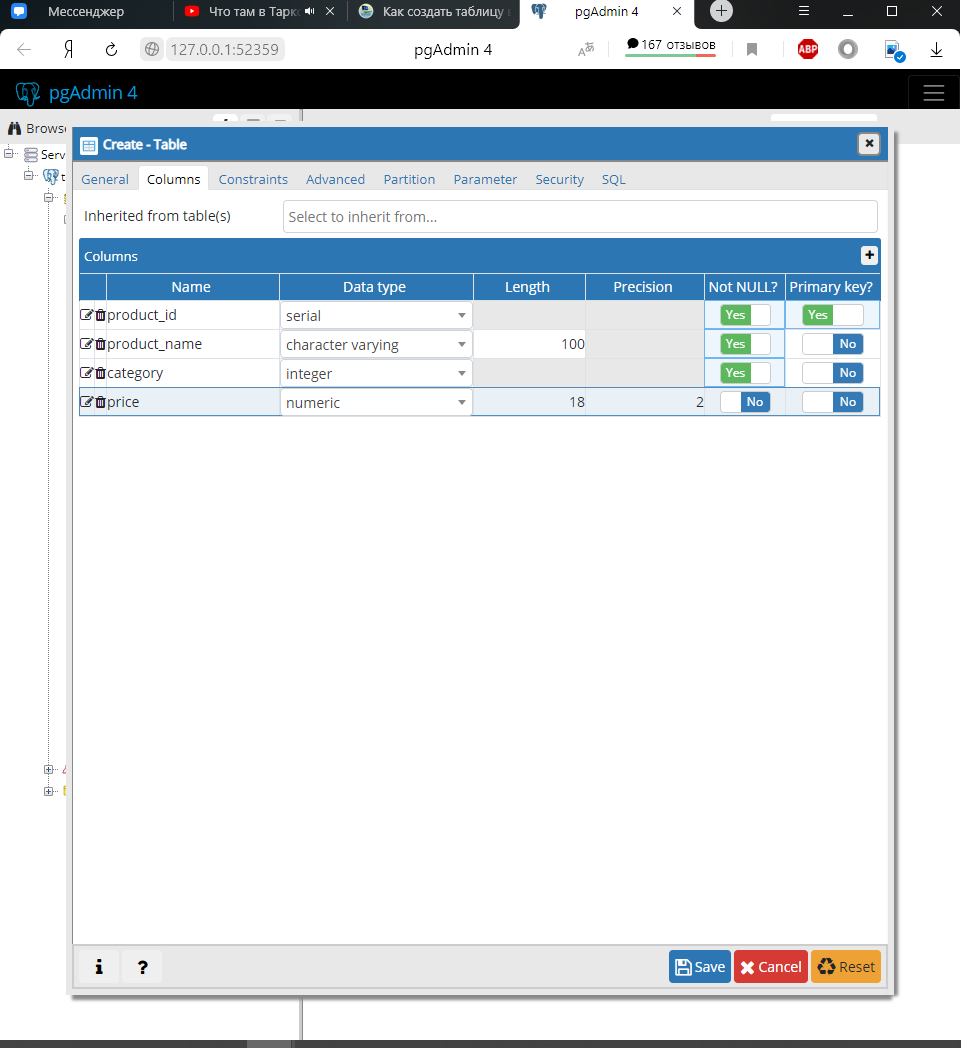
Далее точно так же добавляем столбец для хранения наименования категории. При этом тип данных у нас уже должен быть текстовый, например, character varying (VARCHAR) с длинной 100. Он уже не должен быть первичным ключом и значения генерировать здесь не нужно.

После ввода всех данных мы можем сохранить все изменения, нажав на кнопку «Сохранить», тем самым создав таблицу.

Затем точно также создаем таблицу с товарами и определяем соответствующие для нее столбцы. Столбец идентификатора товара определяем, как первичный ключ, и включаем у него идентификацию с автоматическим генерированием значений.

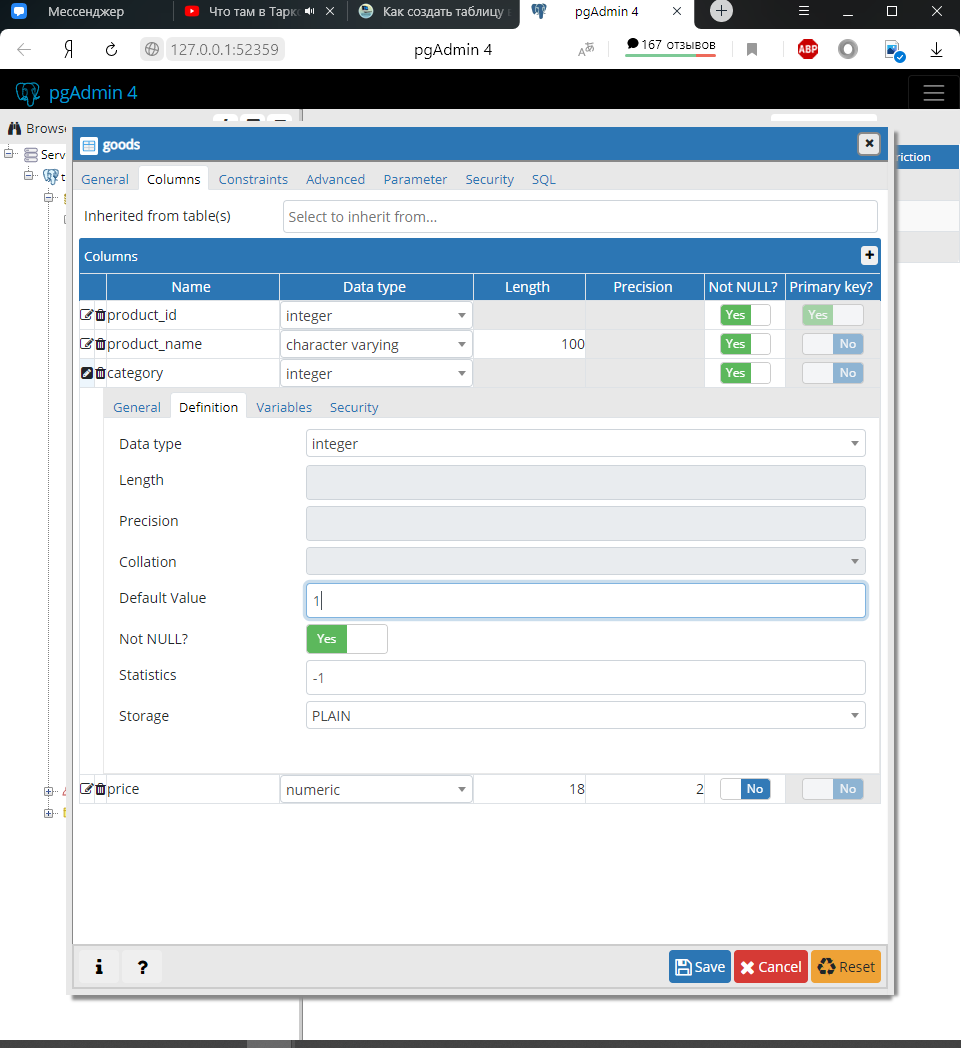


Затем точно также создаем таблицу с товарами и определяем соответствующие для нее столбцы. Столбец идентификатора товара определяем, как первичный ключ, и включаем у него идентификацию с автоматическим генерированием значений.



После того как столбцы определены, нам необходимо добавить значение по умолчанию для столбца category, а также определить ограничение внешнего ключа.

Чтобы у столбца задать значение по умолчанию, необходимо открыть детализированные свойства столбца, перейти там на вкладку «Ограничения» и в поле «По умолчанию» указать значение, которое будет присваиваться по умолчанию, например, 1.



Осталось определить ограничение внешнего ключа, иными словами, чтобы столбец category таблицы goods ссылался на столбец category\_id таблицы categories, таким образом, мы определим связь между этими таблицами.

Чтобы это сделать, переходим на вкладку «Ограничения» (основной формы создания таблиц), затем переходим на вкладку «Внешний ключ» и с помощью кнопки плюс «+» добавляем новый внешний ключ.

Вводим название ограничения, и в детализированных свойствах на вкладке «Столбцы» задаем связь между таблицами, т.е. указываем столбцы и нажимаем на плюс «+» для добавления связи.

После этого все требования, указанные в нашей задаче, будут выполнены и мы можем нажать кнопку «Сохранить» для создания таблицы.

