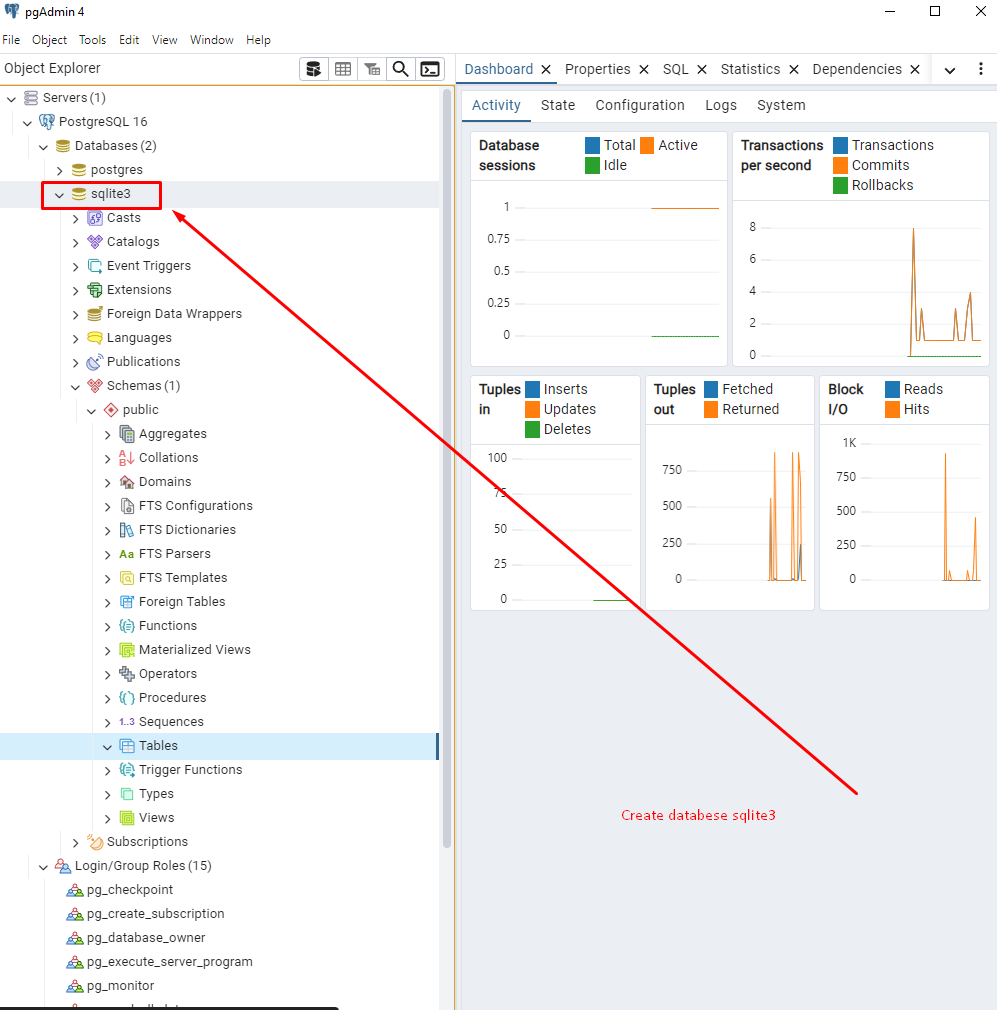
**Домашнее задание по теме "Настраиваем СУБД postgre в django"**

**Задание:**

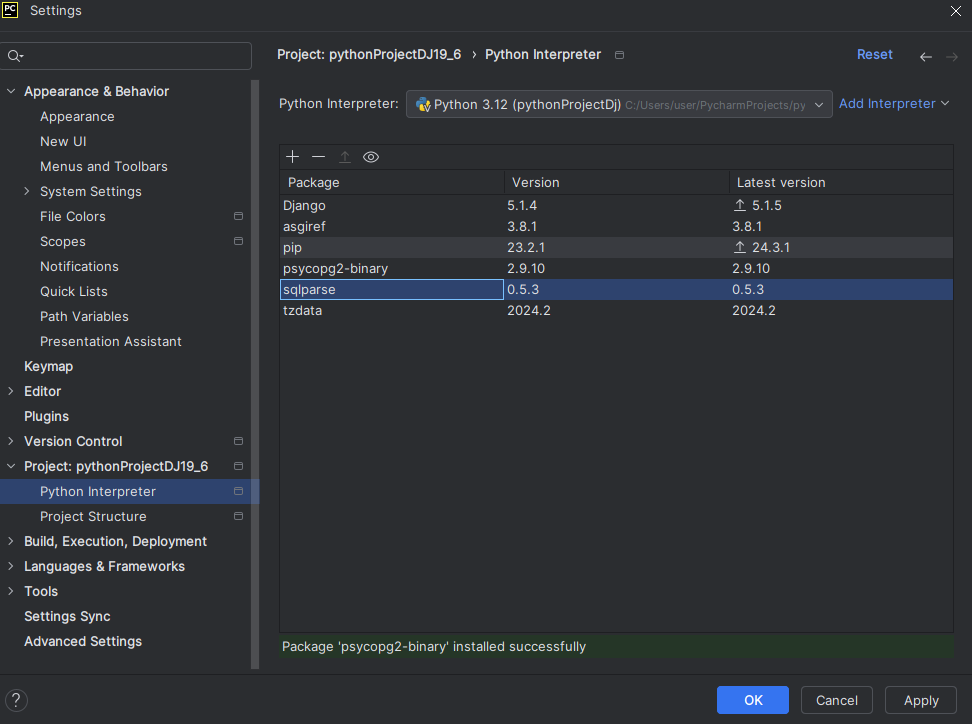
**Часть 1:**

1) **Старт работы с PostgreSQL.** Скачайте и установите официальный пакет pgsql для  
вашей ОС. Установите PGadmin и подключитесь к локальному серверу, создайте новую  
базу данных и прикрепите скриншот выполненного задания.  
2) **В своем Django проекте** установите драйвер psycopg2 для работы с базой PGSQL  
используя менеджер пакетов pip. В файле настроек проекта выполните подключение к  
базе данных:  
DATABASES = {  
'default': {  
'ENGINE': 'django.db.backends.postgresql',  
'NAME': 'имя\_базы\_данных',  
'USER': 'имя\_пользователя',  
'PASSWORD': 'пароль',  
'HOST': '127.0.0.1',  
'PORT': '5432',  
}  
}  
  
**Создайте и выполните** миграции используя makemigrations и migrate. Проверьте  
подключение, в базе данных должны создаться технические таблицы Django проекта.  
3) **Перенесите** любую созданную ранее модель базы данных в PGSQL. Используйте  
Джанго модели и миграции. Создайте несколько таблиц через конструктор PGadmin и  
свяжите их с Джанго проектом, создав необходимые модели.

Создали пустую базу



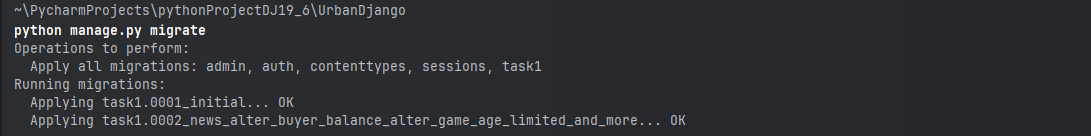
Ставим драйвер



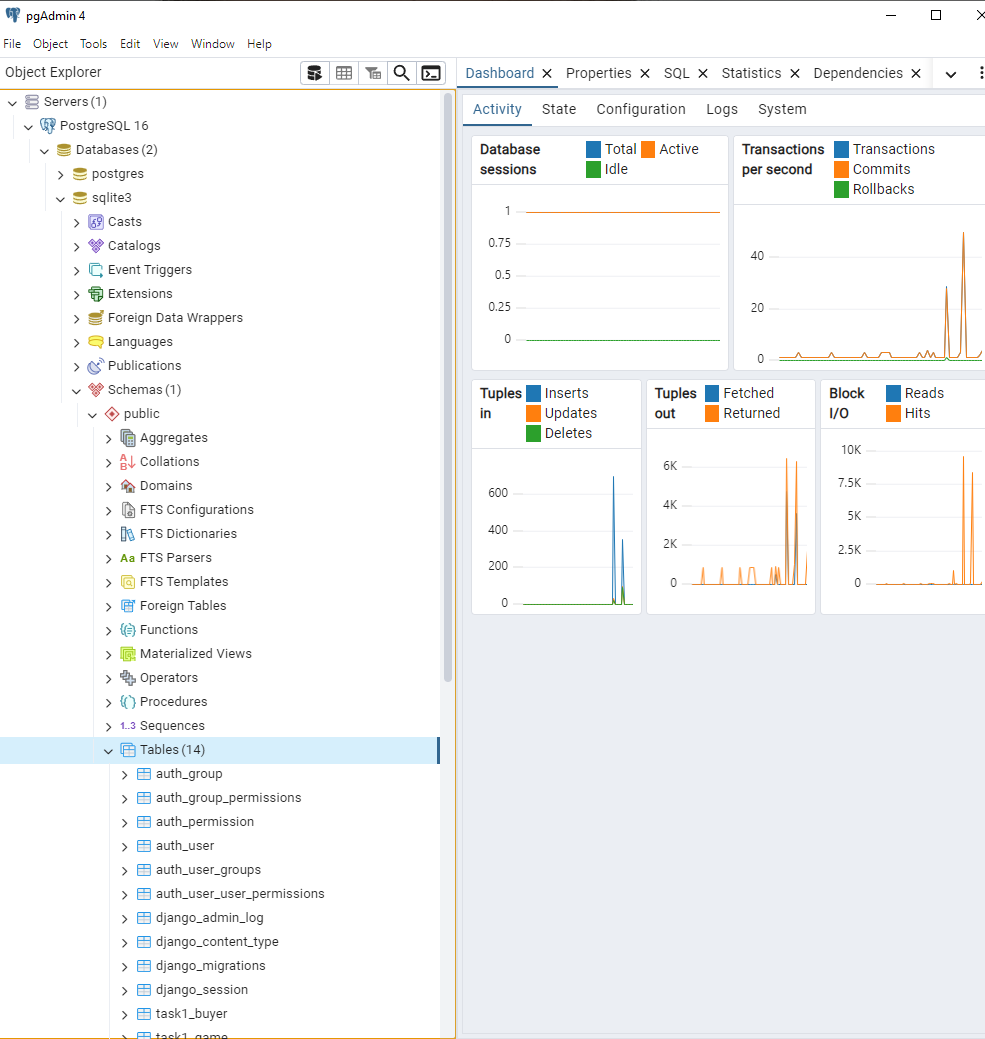
python manage.py makemigrations



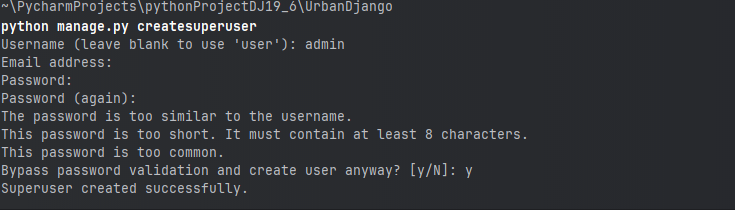
python manage.py migrate



Добавились базы с предыдущего проекта.



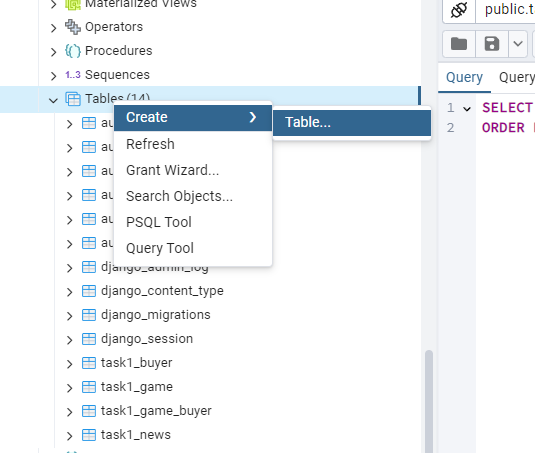
python manage.py createsuperuser

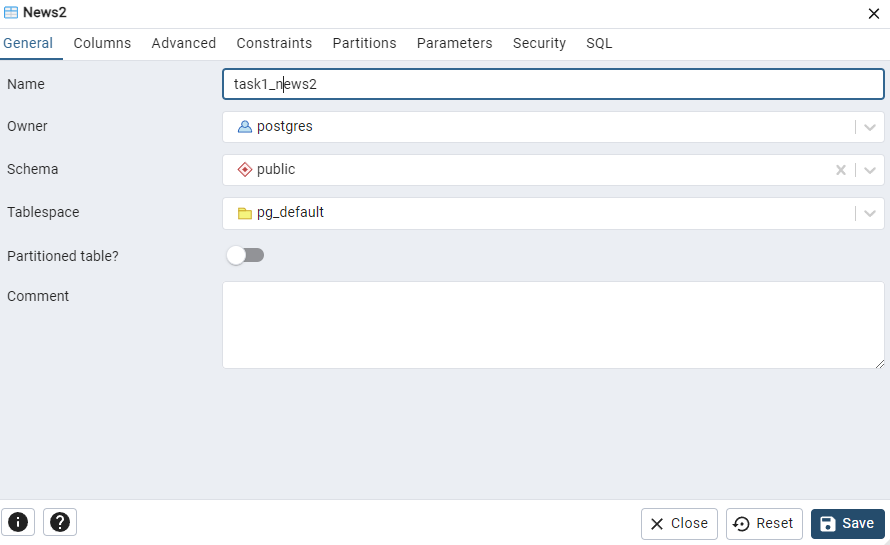


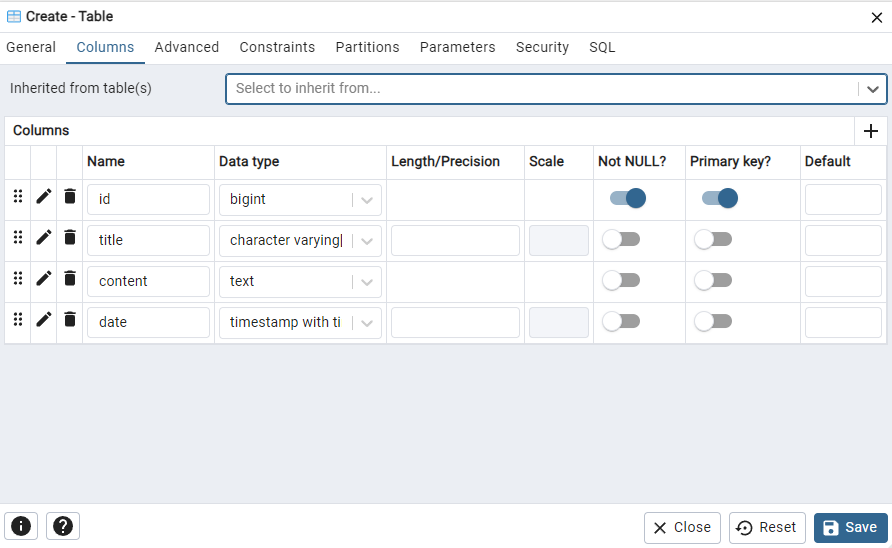
Создайте несколько таблиц через конструктор PGadmin и  
свяжите их с Джанго проектом, создав необходимые модели.

Так как нет смысла усложнять задачу разберем как это сделать, на примере таблицы News, создадим ее клона task1\_news2 в нашей базе данных и пропишем ее в модели в файл models.py как News2. И можем делать те же запросы что и для первой таблицы, ничем не будет отличаться.

Заходим в PGadmin выбираем раздел таблицы и создаем таблицу с аналогичным именем

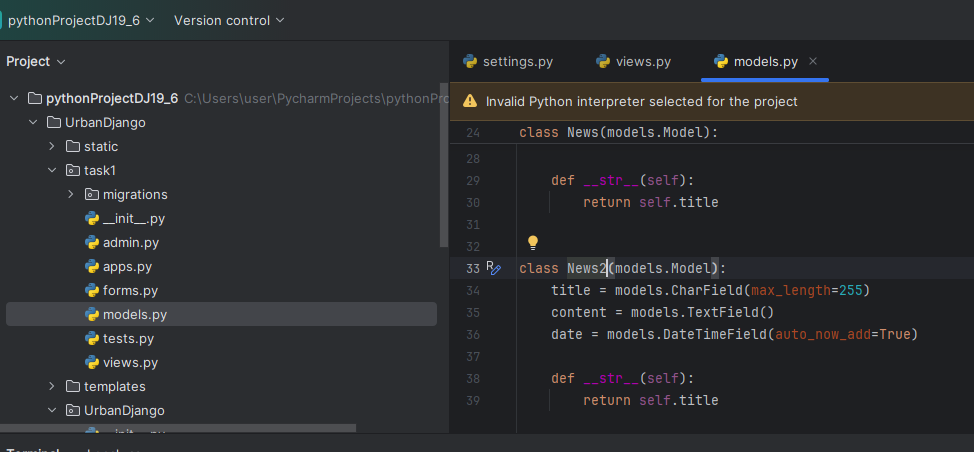






**Нажимаем Save**

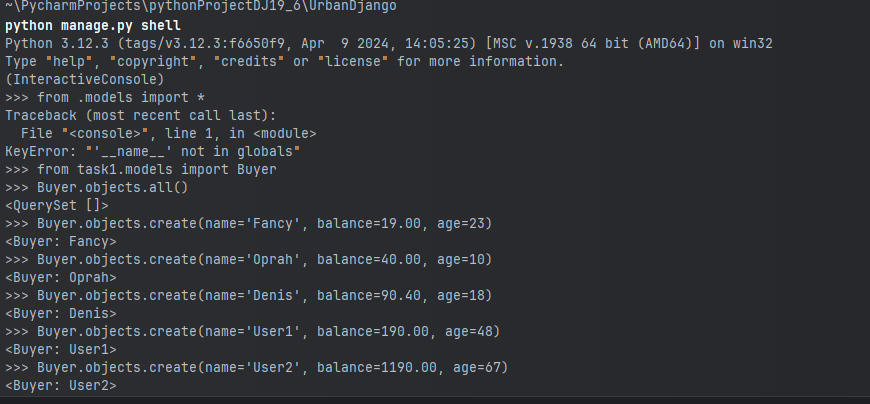
**И добавляем в Джанго**



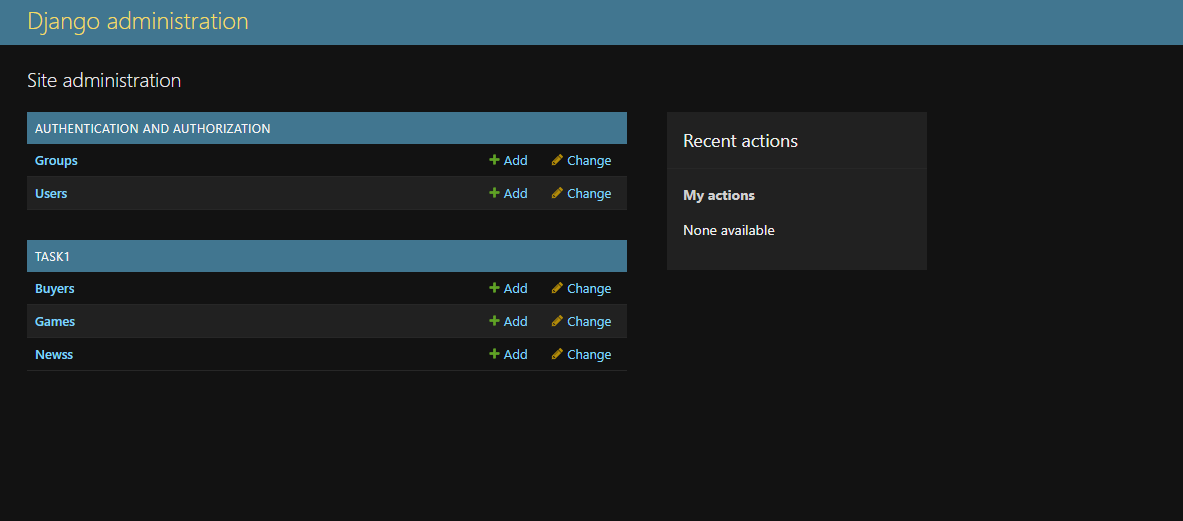
**Часть 2:**

4) **Используйте** Django ORM для тестирования запросов в вашу базу данных.  
Выполните команду Python manage.py shell. Импортируйте необходимые модели из вашей  
базы данных и создайте не менее 4 запросов, например:  
Запрос на получение всех объектов базы данных и конкретного по id.  
Запрос на фильтрацию  
Запрос на добавление или удаление объекта

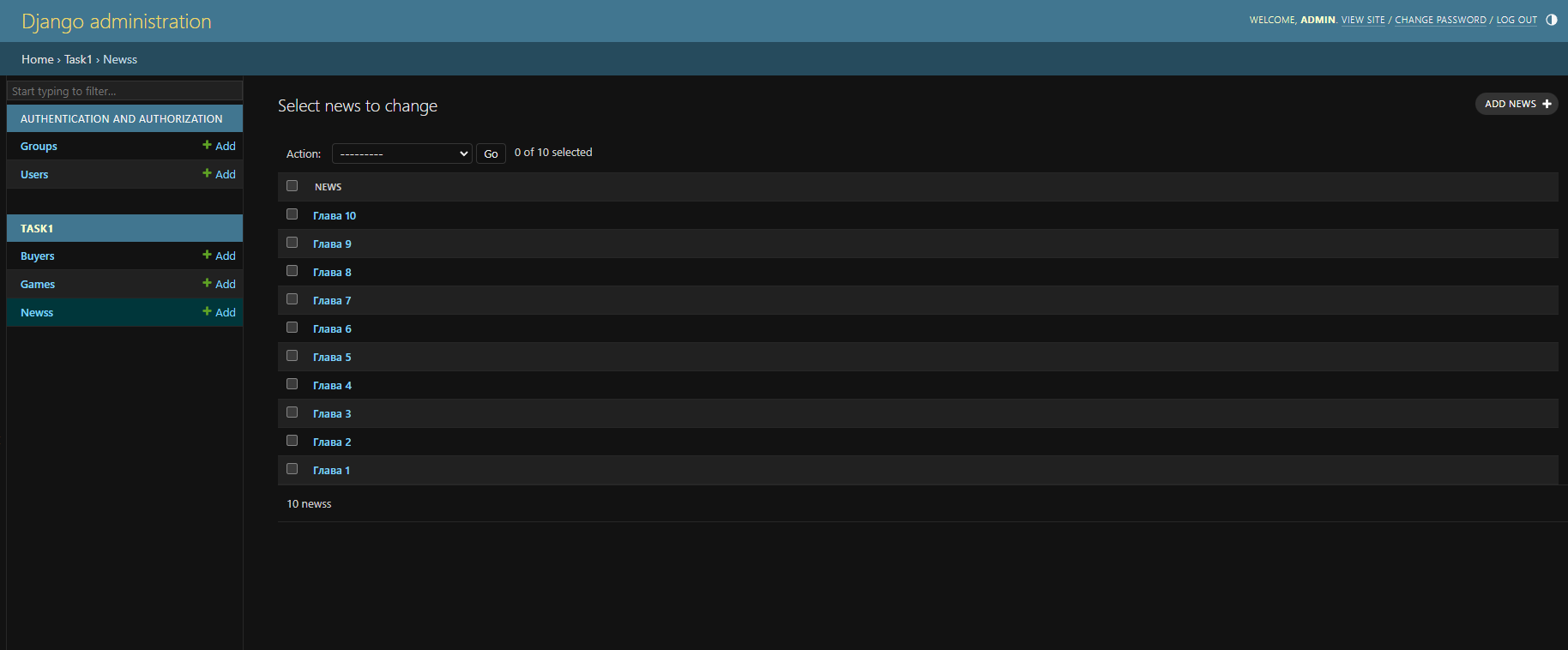
Добавим пользователей в пустую базу



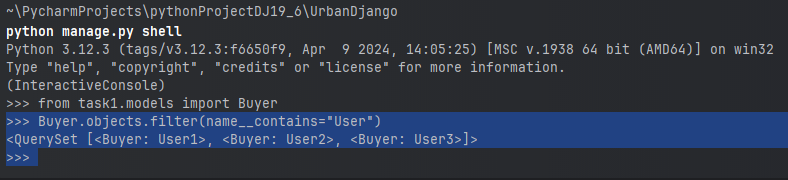
Переходим в админку



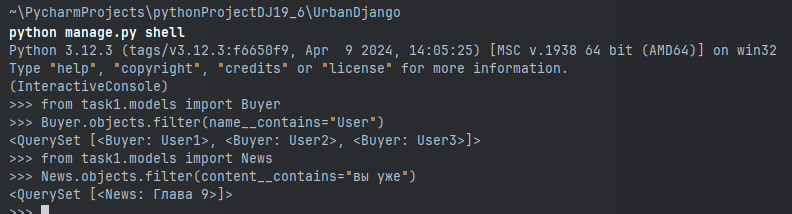
Проверяем работу с базой через админку. В таблицу News успешно внесены данные.



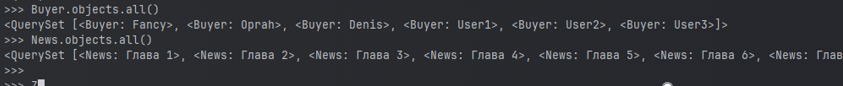
После внесения данных продолжаем тестировать запросы в базу данных выведем всех пользователей, содержащих в имени User



Импортируем модель News и также просим показать где встречается сочетание «вы же»



Вывести в консоль содержимое заполненных таблиц



Выводим сколько раз встречается пользователи, содержащие в имени User

