

Копирование и настройка образа сетевого сервера и сопутствующего ПО

Копирование образа:

1. Взять карту памяти, сбросить на ней файловую систему. Например при помощи "Управление дисками" в Windows
2. Скопировать образ:

```
sudo dd if=<путь до файла, его название>.img of=<путь до карты памяти> bs=1M status=progress
```

При необходимости уменьшить параметр bs

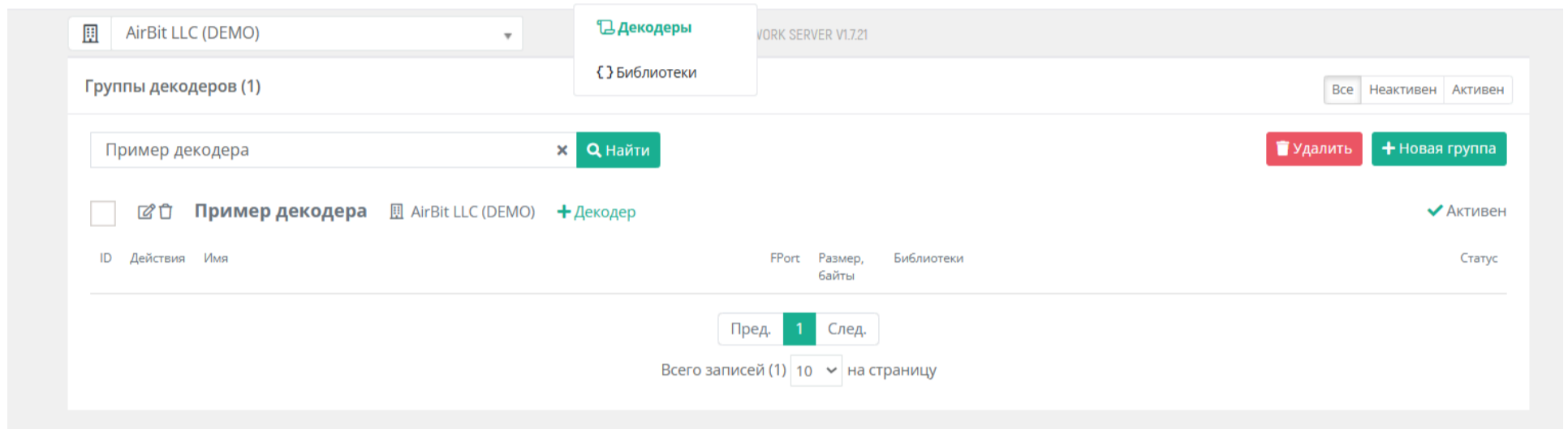
Настройка декодеров и отправки данных

1. Для создания декодеров необходимо перейти в раздел "Декодеры" подраздел "декодеры". Декодеры пишутся на языке JavaScript

▼ Подраздел "Декодеры"

Для создания нового декодера необходимо нажать "+ Новая группа" в правом верхнем углу подраздела

Обзор Шлюзы Устройства Настройки </> Декодеры Пользователи Отчеты >_ Логи

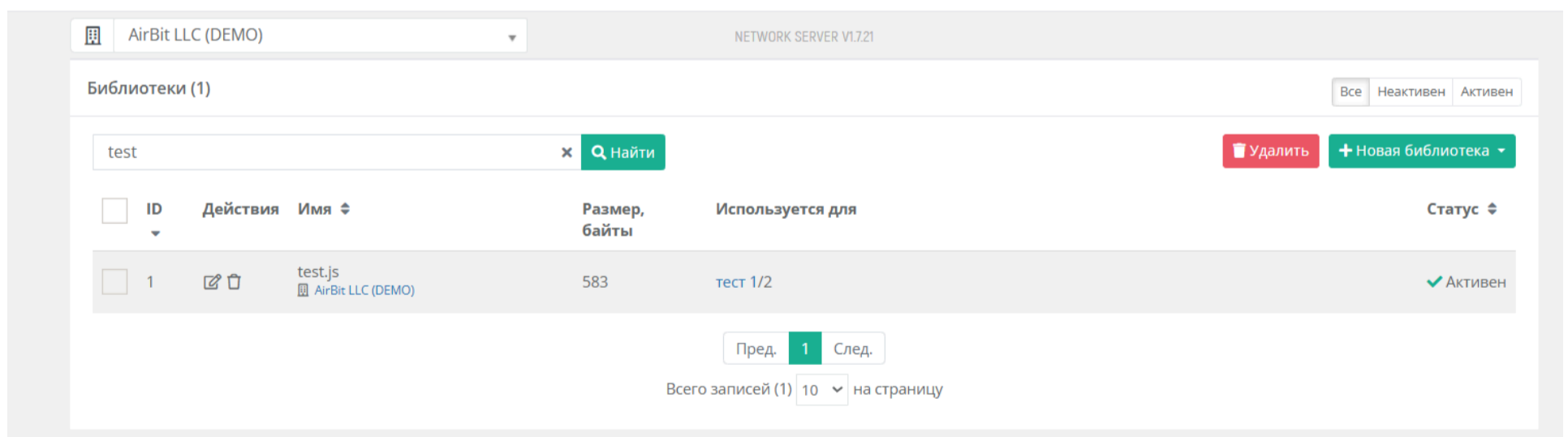


Если внутри нескольких декодеров часто используются одинаковые функции, то их можно вынести в отдельную библиотеку

▼ Подраздел "Библиотеки"

Для создания новой библиотеки необходимо нажать "+ Новая библиотека" в правом верхнем углу подраздела.

Обзор Шлюзы Устройства Настройки </> Декодеры Пользователи Отчеты >_ Логи



2. Для отправки данных с устройства необходимо при его заведении добавить в одну из стандартных групп "Grafana", также при добавлении устройства задает декодер.

▼ Пример

Для более подробного описания добавления устройства смотри инструкцию по "AirBit LoRaWAN Network Server"

Профиль	-- Не выбрано --
DevEUI *	DevEUI <small>Введите 8-ми байтную (16-ти символьную) строку в HEX формате</small>
Имя *	Имя
Класс *	Класс А
Активация *	ОТАА
AppEUI *	AppEUI <small>Введите 8-ми байтную (16-ти символьную) строку в HEX формате</small>
AppKey *	AppKey <small>Введите 16-и байтную (32-х символьную) строку в HEX формате</small>
Сеть	-- Не выбрано -- <small>Если не выбрать, то будет использована сеть по умолчанию</small>
Частотный план	-- Не выбрано -- <small>Если не выбрать, то будет использован частотный план по умолчанию</small>
Группы	Grafana x
Декодер	Пример декодера x
Интервал передачи данных	1 сутки <small>Выберите наиболее подходящий интервал передачи данных</small>

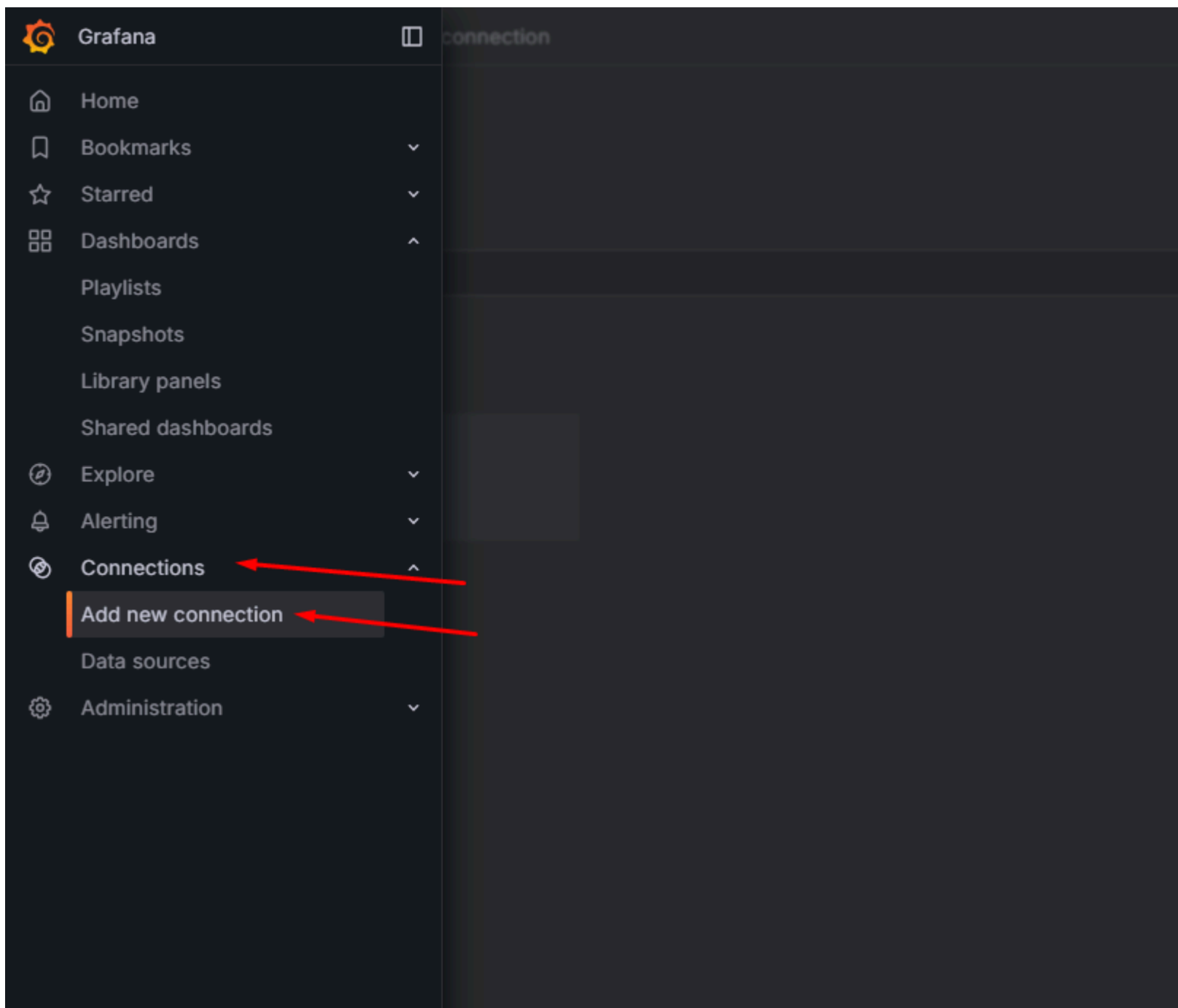
Настройка получения данных от устройств в Grafana

1. Зайти и авторизоваться в Grafana: `<ip адрес локальной установки>:3000`

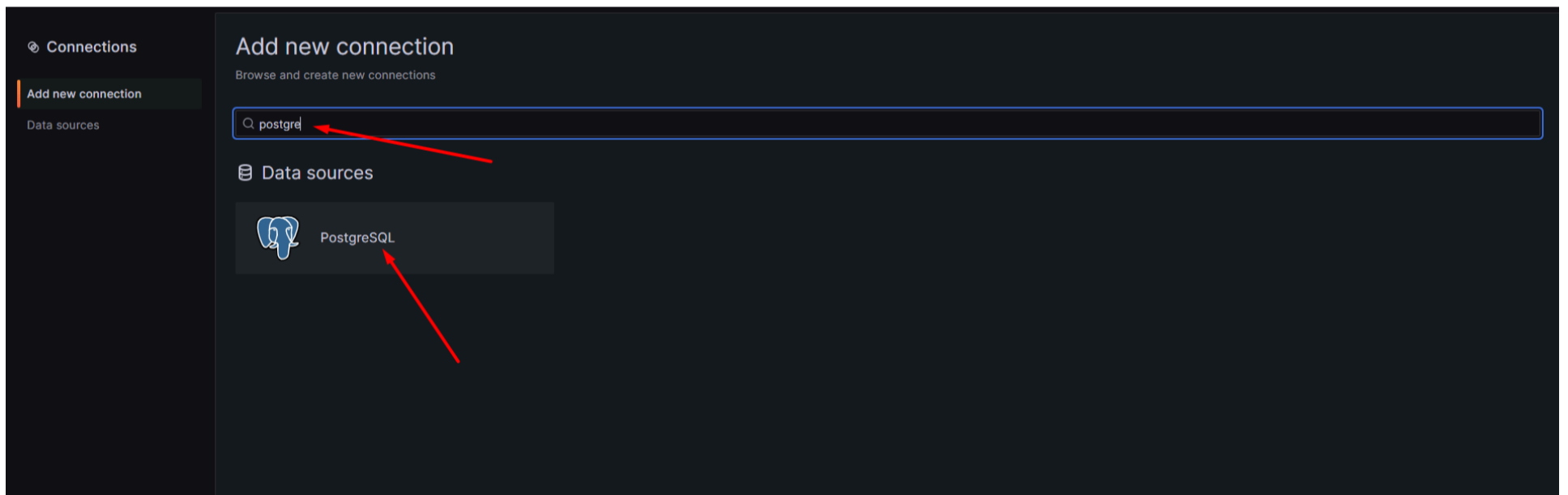
По умолчанию логин:airpi и пароль: 97HJhd1IsFF

2. Добавление поддержки PostgreSQL

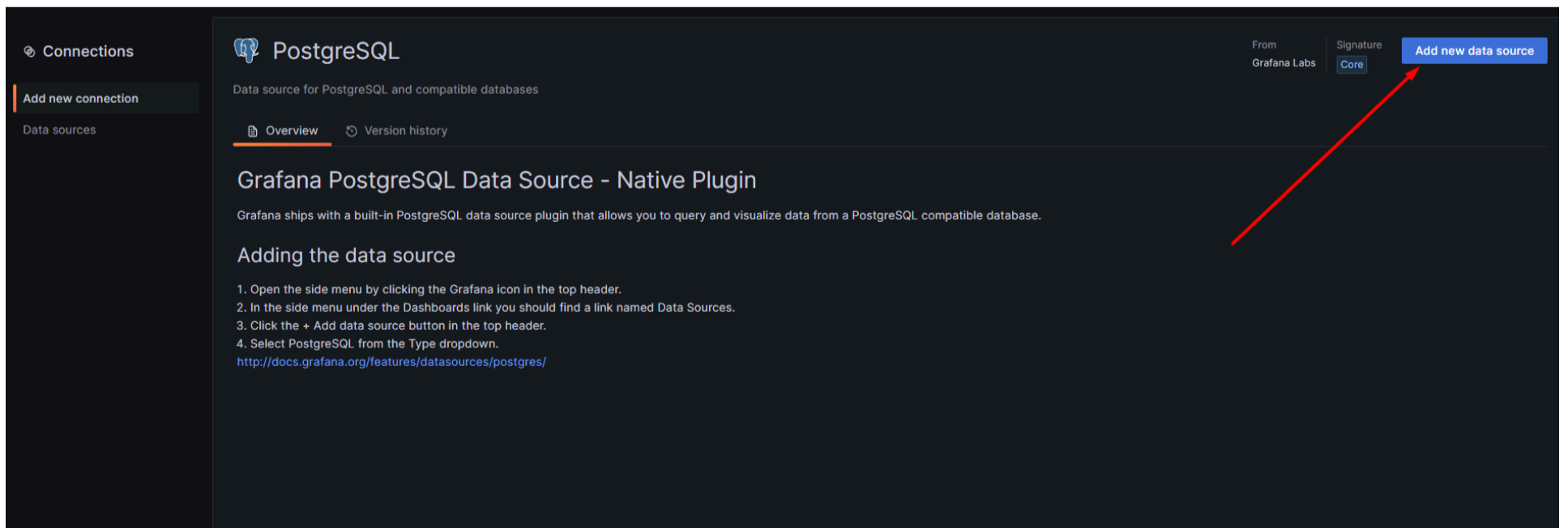
▼ Шаг 1



▼ Шаг 2

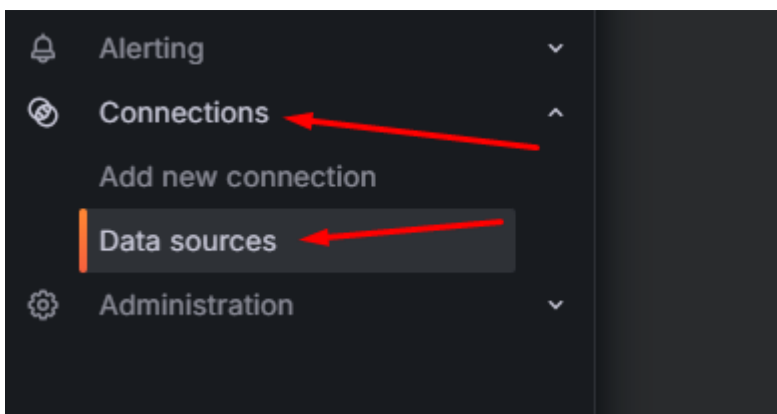


▼ Шаг 3

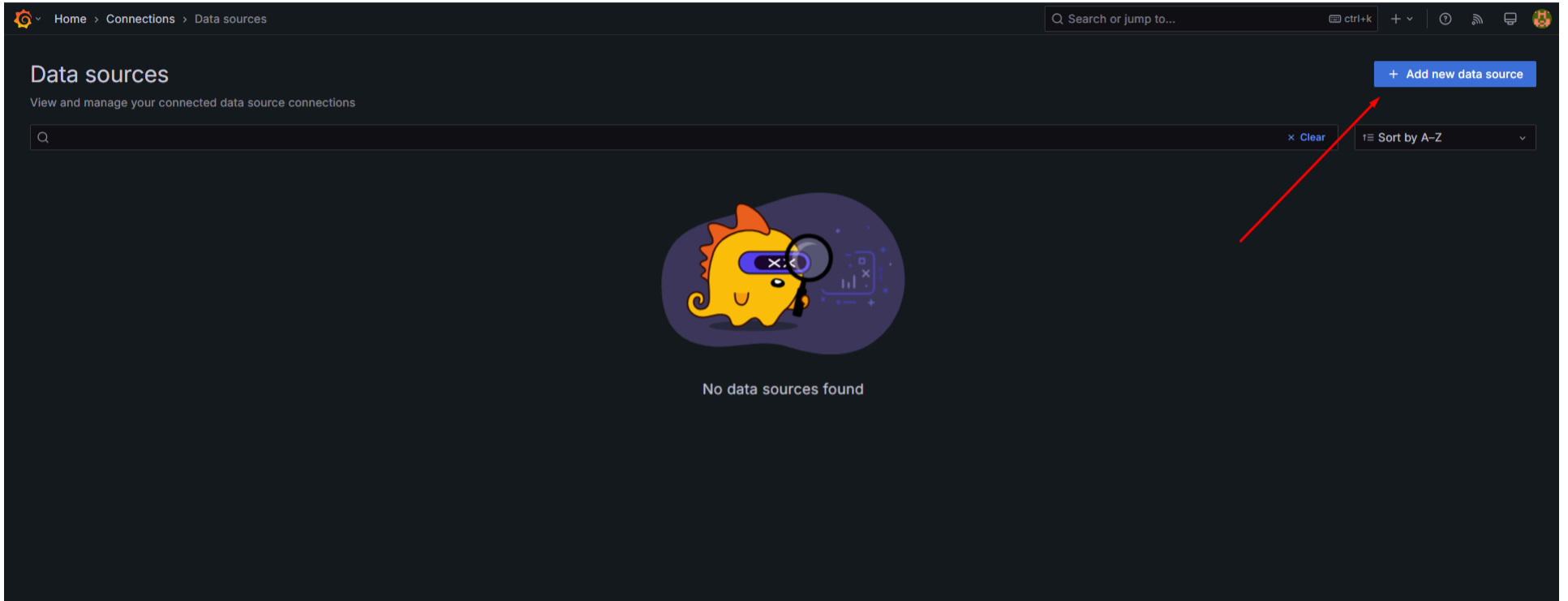


3. Добавить новый источник

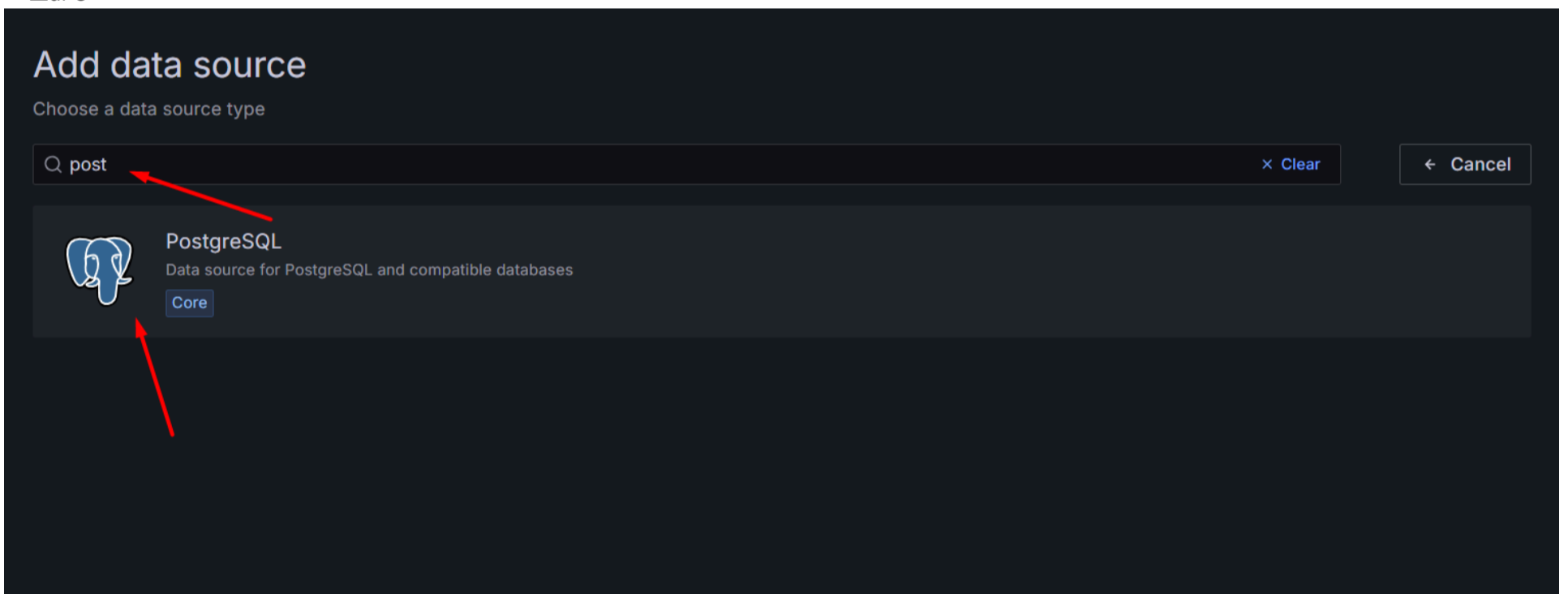
▼ Шаг 1



Шаг 2



Шаг 3



Шаг 4

Название БД с параметрами: "parameters"

По умолчанию логин: airbit и пароль: 97HJhd1IsFF

The image shows the Grafana configuration page for a PostgreSQL data source. The page is titled "grafana-postgresql-datasource" and includes navigation links for "Type", "Alerting", "Explore data", and "Build a dashboard". The "Type" is set to "PostgreSQL" and "Alerting" is marked as "Supported".

Settings

Name: grafana-postgresql-datasource (Default)

Before you can use the Postgres data source, you must configure it below or in the config file. For detailed instructions, [view the documentation](#).

Fields marked with * are required

User Permissions

The database user should only be granted SELECT permissions on the specified database & tables you want to query. Grafana does not validate that queries are safe so queries can contain any SQL statement. For example, statements like `DELETE FROM user;` and `DROP TABLE user;` would be executed. To protect against this we Highly recommend you create a specific PostgreSQL user with restricted permissions. Check out the docs for more information.

Connection

Host URL *
host.docker.internal

Database name *
parametrs

Authentication

Username *
airbit

Password *
configured

Username *
airbit

Password *
configured

TLS/SSL Mode
disable

Additional settings

PostgreSQL Options

Version
15

Min time interval
1m

TimescaleDB

Connection limits

Max open
100

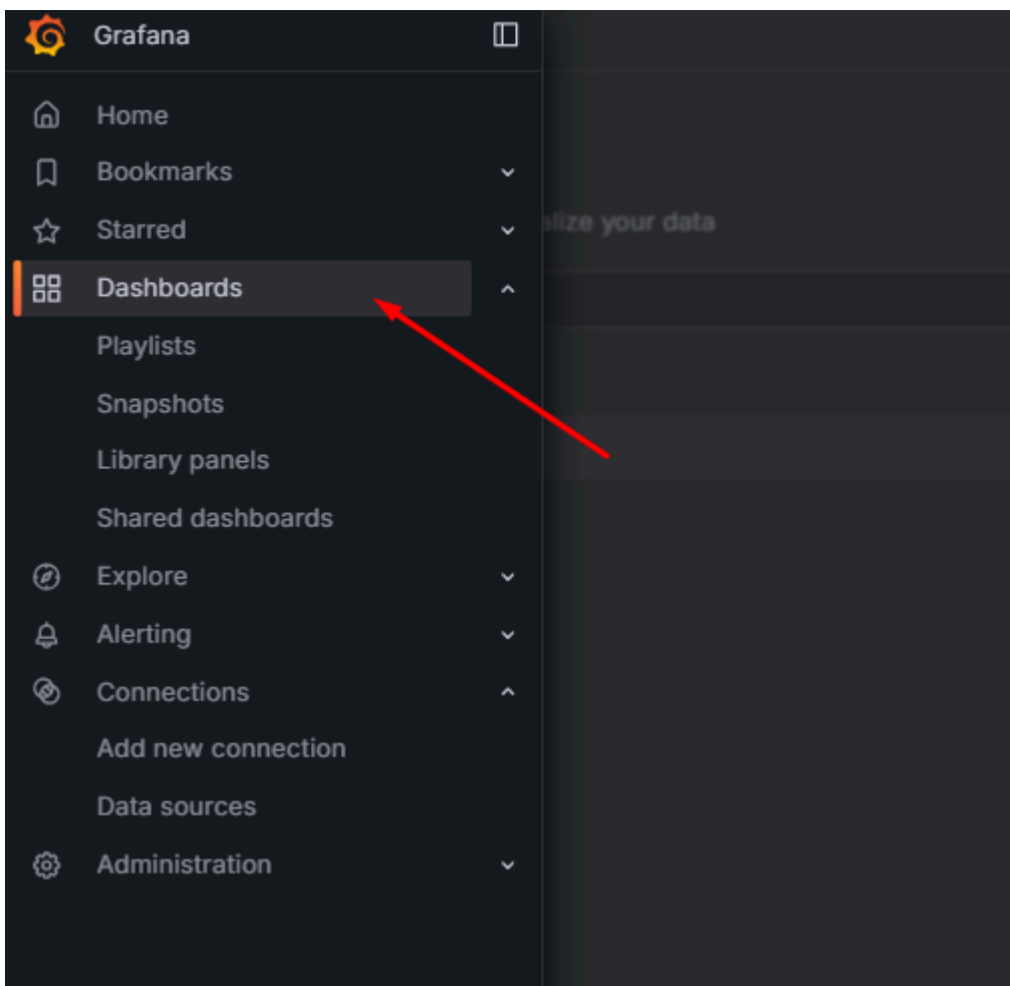
Auto max idle

Max idle
100

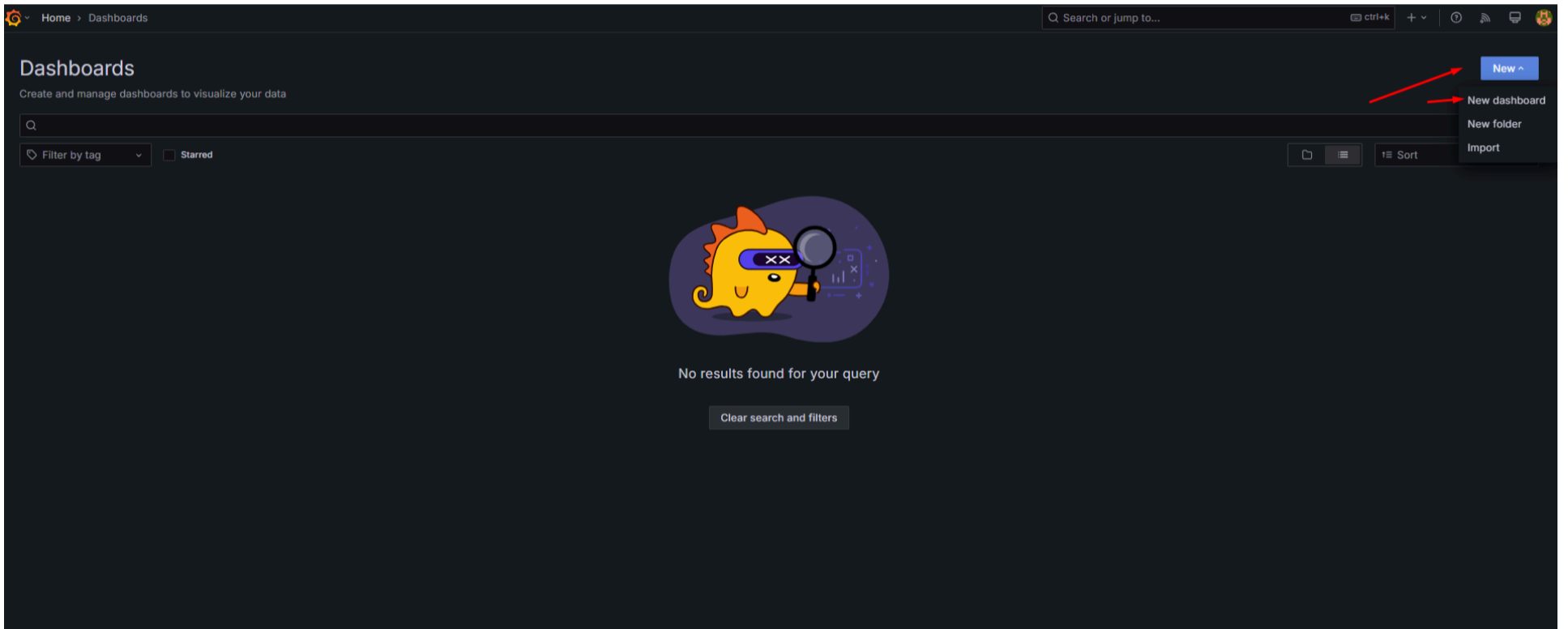
Max lifetime
14400

5. Пример настройки виджета

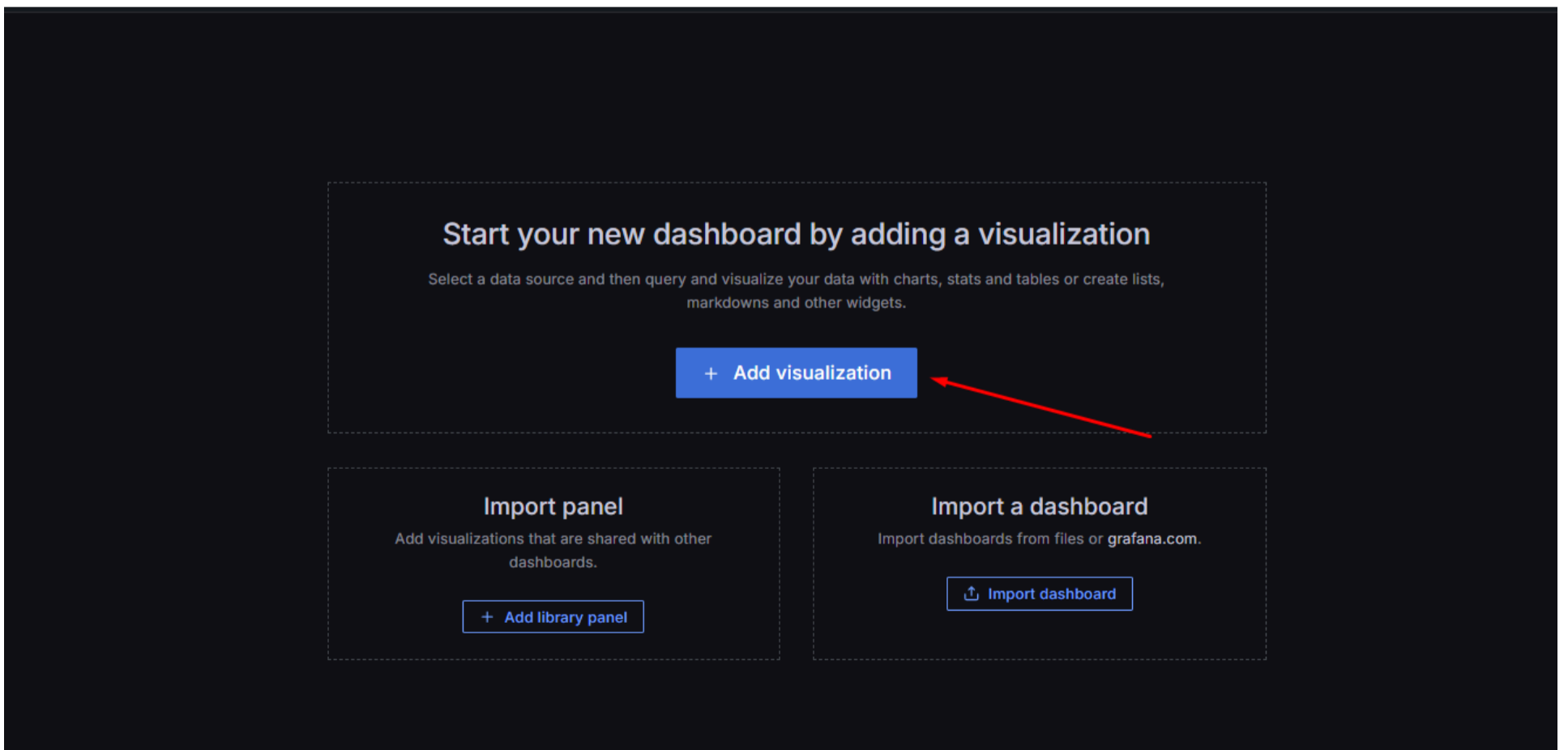
▼ Шаг 1



▼ Шаг 2

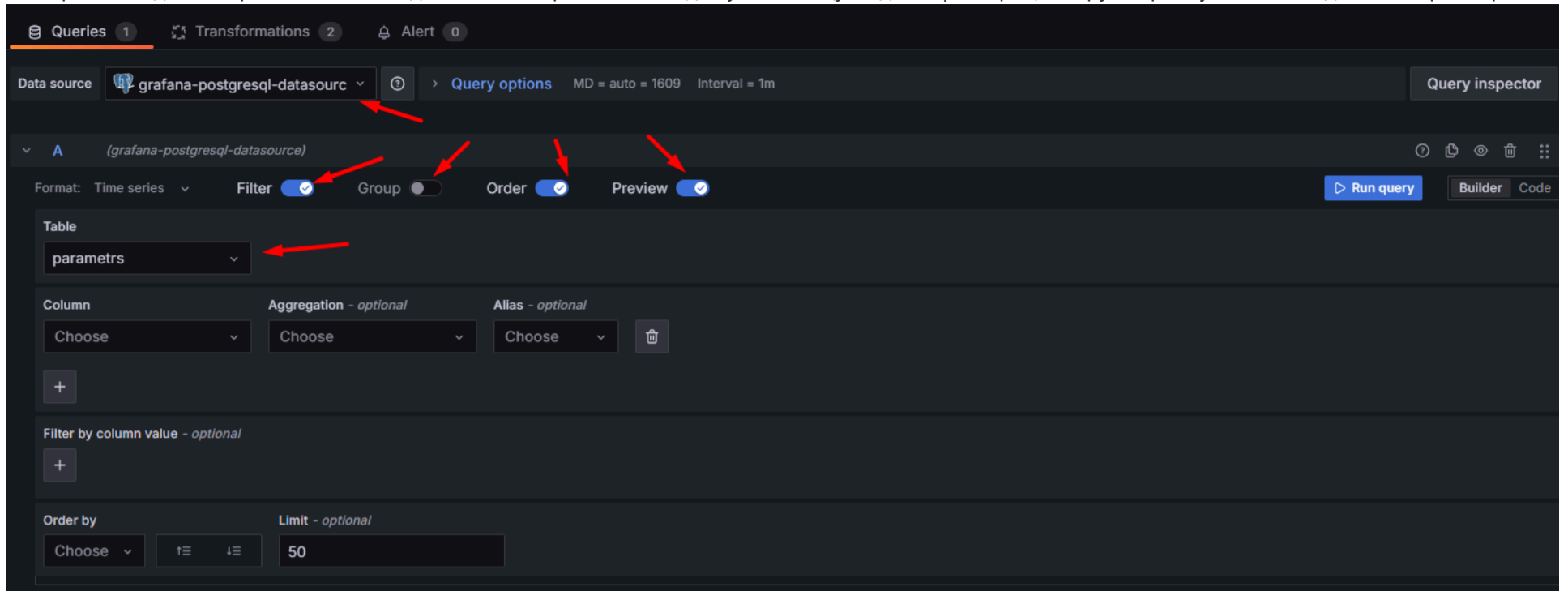


▼ Шаг 3

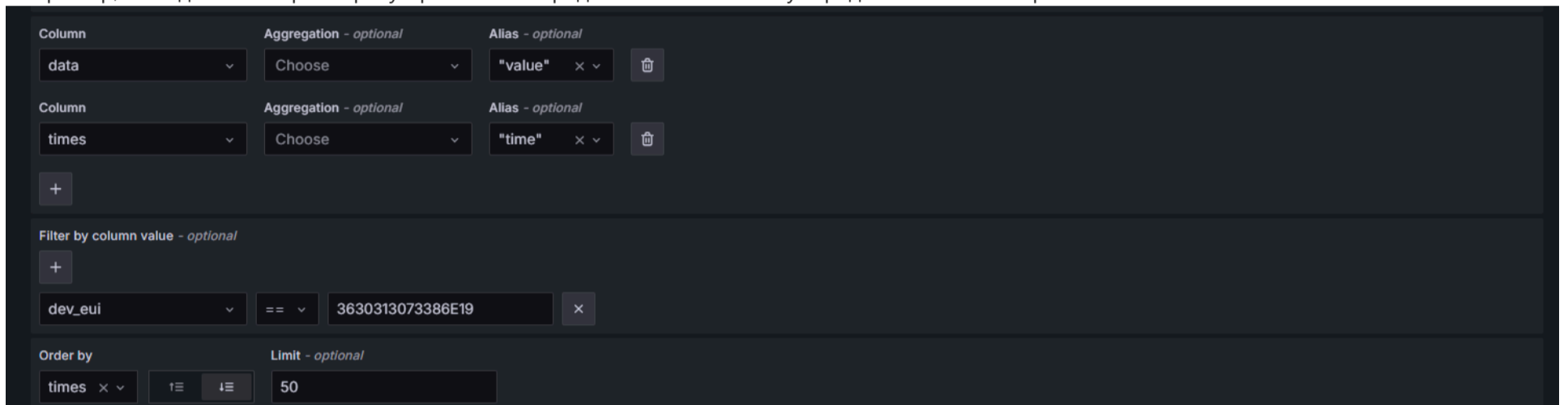


▼ Шаг 4

Выбираем созданный ранее источник данных, выбираем необходимую таблицу, задаем фильтрацию/группировку по необходимым параметрам



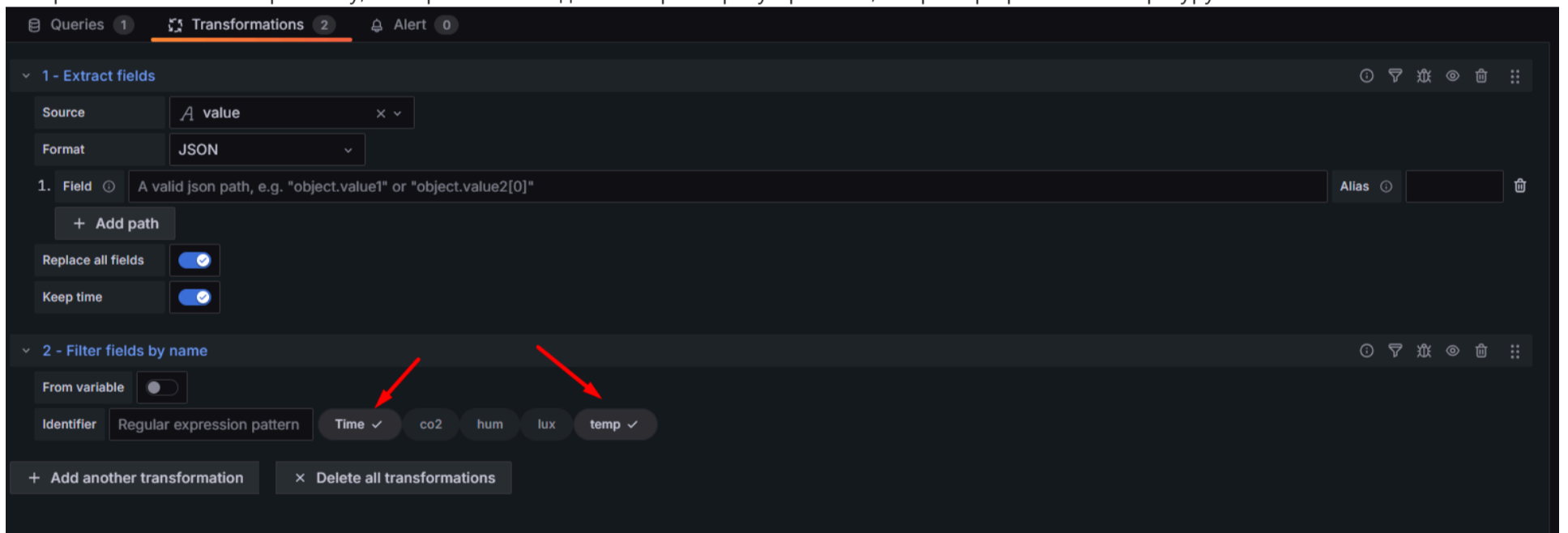
Например, выводим все параметры устройства с определенным DevEUI и упорядочиваем их по времени



Если необходимо вытащить конкретный параметр из JSON, выполняем шаг 5

▼ Шаг 5

Настраиваем согласно скриншоту, выбираем необходимые параметры устройства, например время и температуру



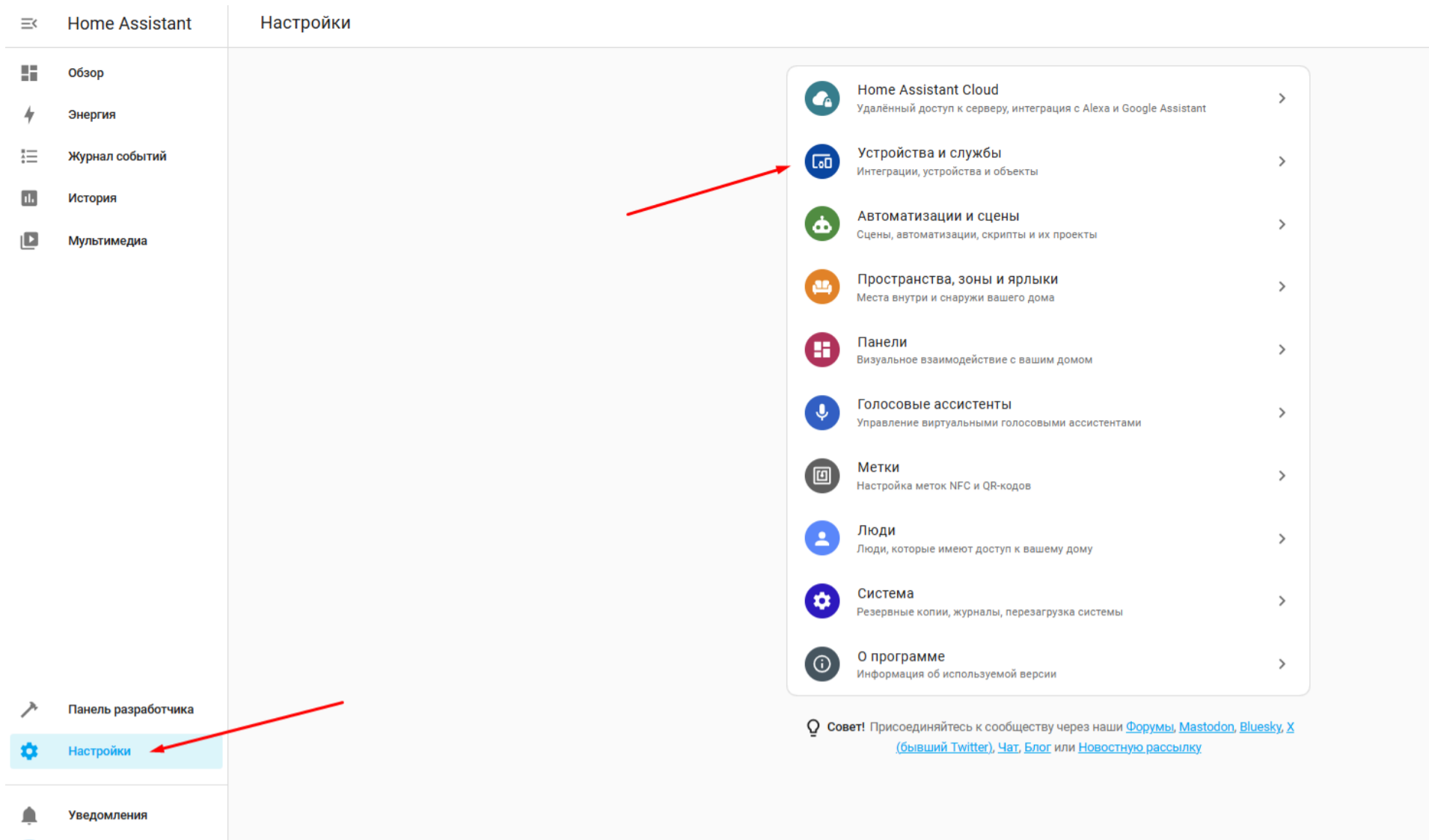
Настройка получения данных от устройств в HomeAssistant

1. Зайти и авторизоваться в HomeAssistant: `<ip адрес локальной установки>:8123`

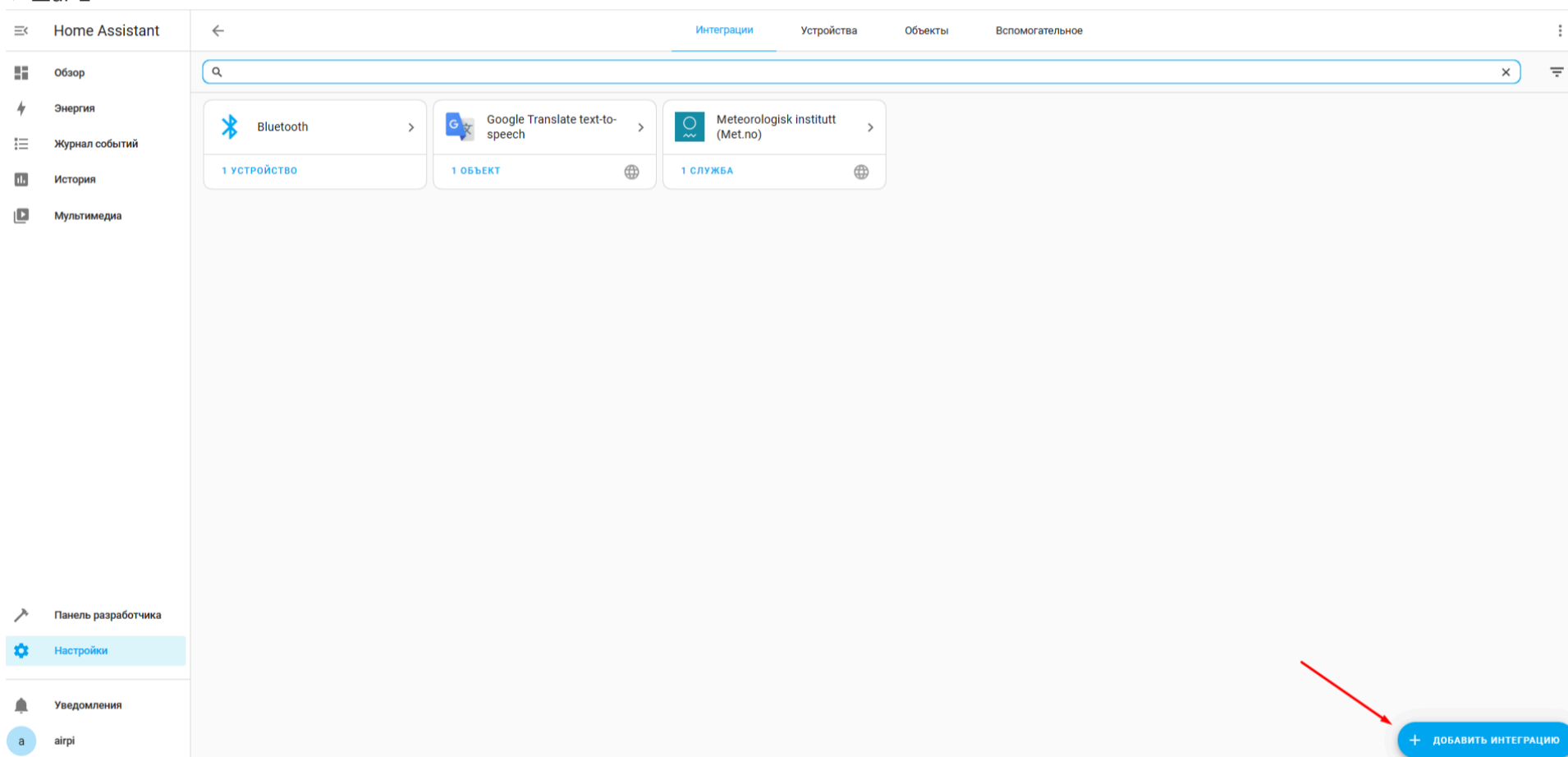
Логин и пароль задаются при первом входе

2. Добавление поддержки MQTT

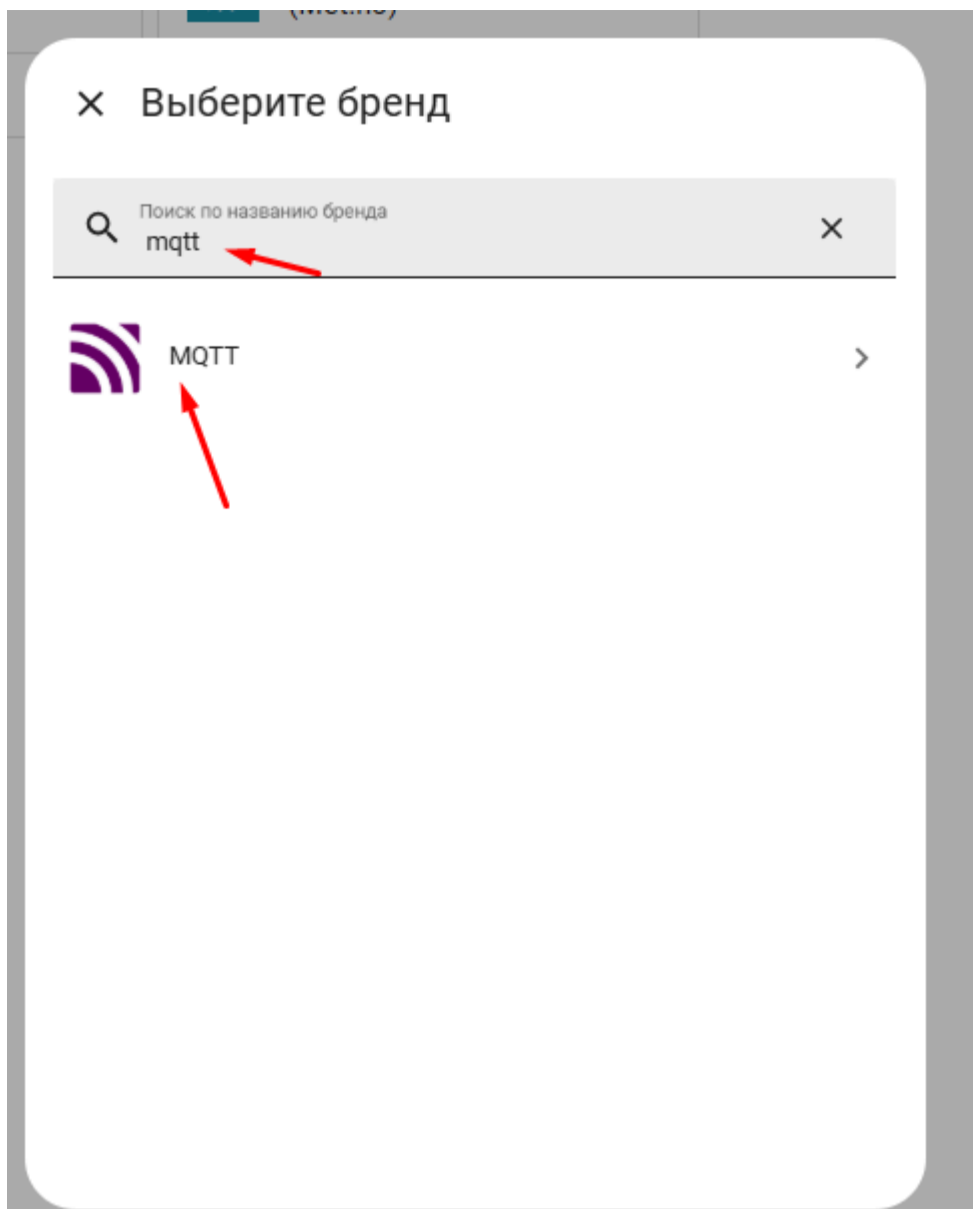
▼ Шаг 1



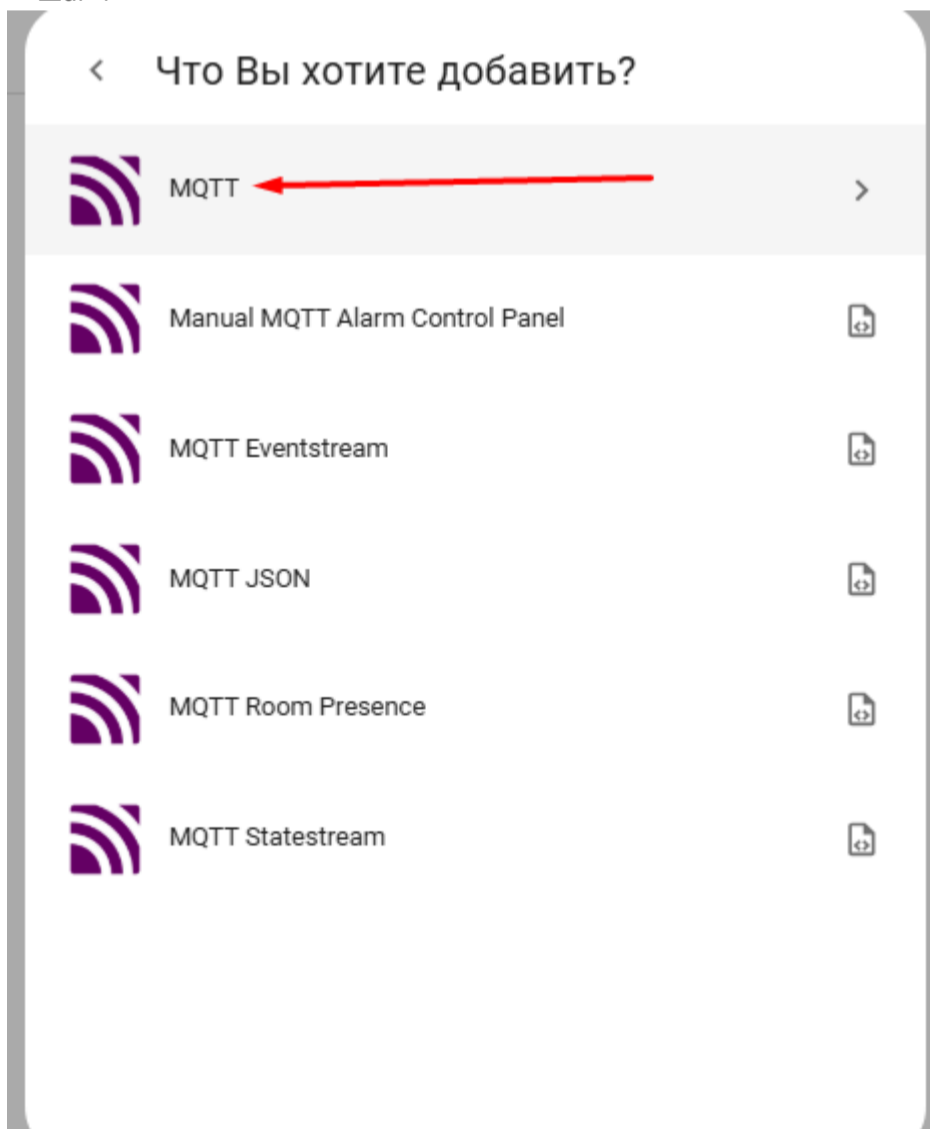
Шаг 2



Шаг 3



▼ Шаг 4



▼ Шаг 5

Логин и пароль аналогичные тем, которые задавали для MQTT брокера

MQTT

Введите информацию о подключении к Вашему брокеру MQTT.

Брокер*
<ip адрес локальной установки>

Имя хоста или IP-адрес вашего брокера MQTT.

Порт*
1883

Порт, который прослушивает ваш брокер MQTT. Например, 1883.

Имя пользователя
Airbit

Имя пользователя для входа в брокер MQTT.

Пароль

Пароль для входа в брокер MQTT.

ПОДТВЕРДИТЬ

▼ Шаг 6

Добавляем в файл конфига MQTT, по умолчанию конфиг находится в директории указанной в пункте 1

```
# Loads default set of integrations. Do not remove.
default_config:

# Load frontend themes from the themes folder
frontend:
  themes: !include_dir_merge_named themes

# Text to speech
tts:
  - platform: google_translate

automation: !include automations.yaml
script: !include scripts.yaml
scene: !include scenes.yaml
modbus: !include modbus.yaml
sensor: !include sensor.yaml
mqtt: !include mqtt.yaml
media_player: !include media_player.yaml
telegram_bot:
```

3. Добавление устройства

▼ Пример конфигурации yaml файла

В той же директории, где находится общий конфиг создаем файл "mqtt.yaml", настраиваем получение параметров из топиков

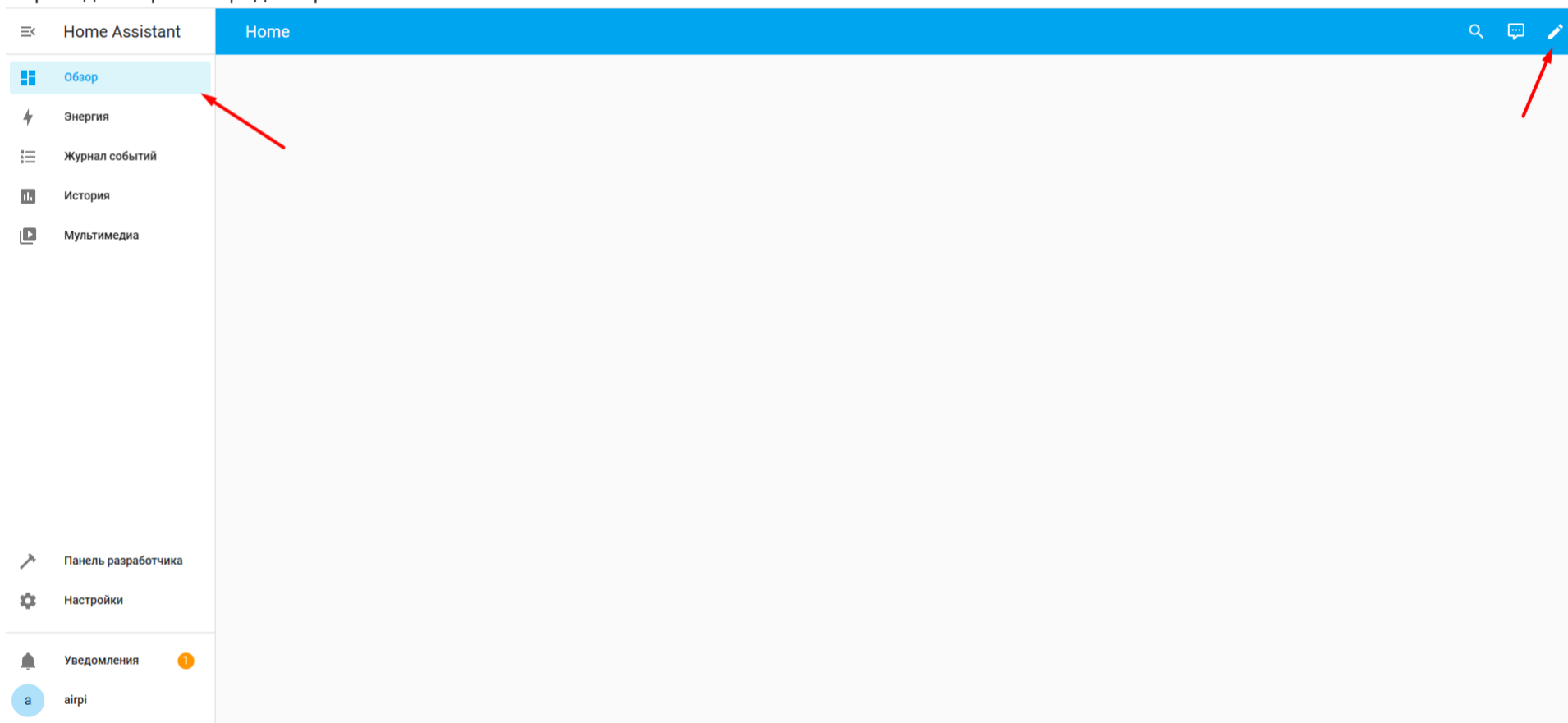
```
sensor:
- name: "Um_temp"
  unique_id: um_temp_id
  state_topic: "ha/3630313073386E19"
  suggested_display_precision: 1
  unit_of_measurement: "°C"
  value_template: "{{value_json.data[0].temp}}".
- name: "Um_lux"
  unique_id: um_lux_id
  state_topic: "ha/3630313073386E19"
  suggested_display_precision: 1
  unit_of_measurement: "lux"
  value_template: "{{value_json.data[0].lux}}"
- name: "Um_co2"
  unique_id: um_co_id
  state_topic: "ha/3630313073386E19"
  suggested_display_precision: 1
  unit_of_measurement: "ppm"
  value_template: "{{value_json.data[0].co2}}"
- name: "Um_hum"
  unique_id: um_hum_id
  state_topic: "ha/3630313073386E19"
  suggested_display_precision: 1
  unit_of_measurement: "%"
  value_template: "{{value_json.data[0].hum}}"
```

Если устройства на сетевом сервере добавлены в группу "Grafana", то данные будут поступать в топики: grafana/{devEUI_устройства}

4. Пример добавления дашборда

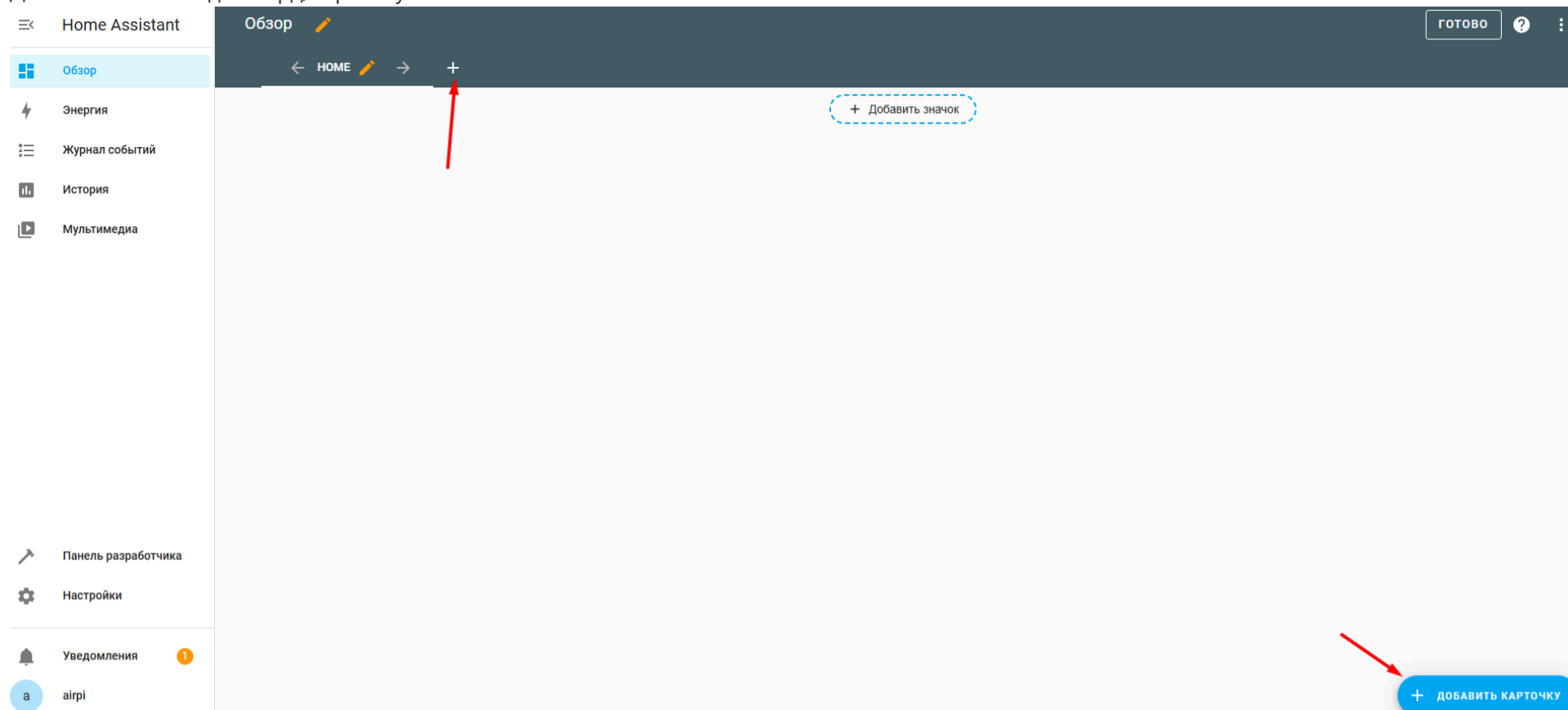
▼ Шаг 1

Переходим в режим редактирования



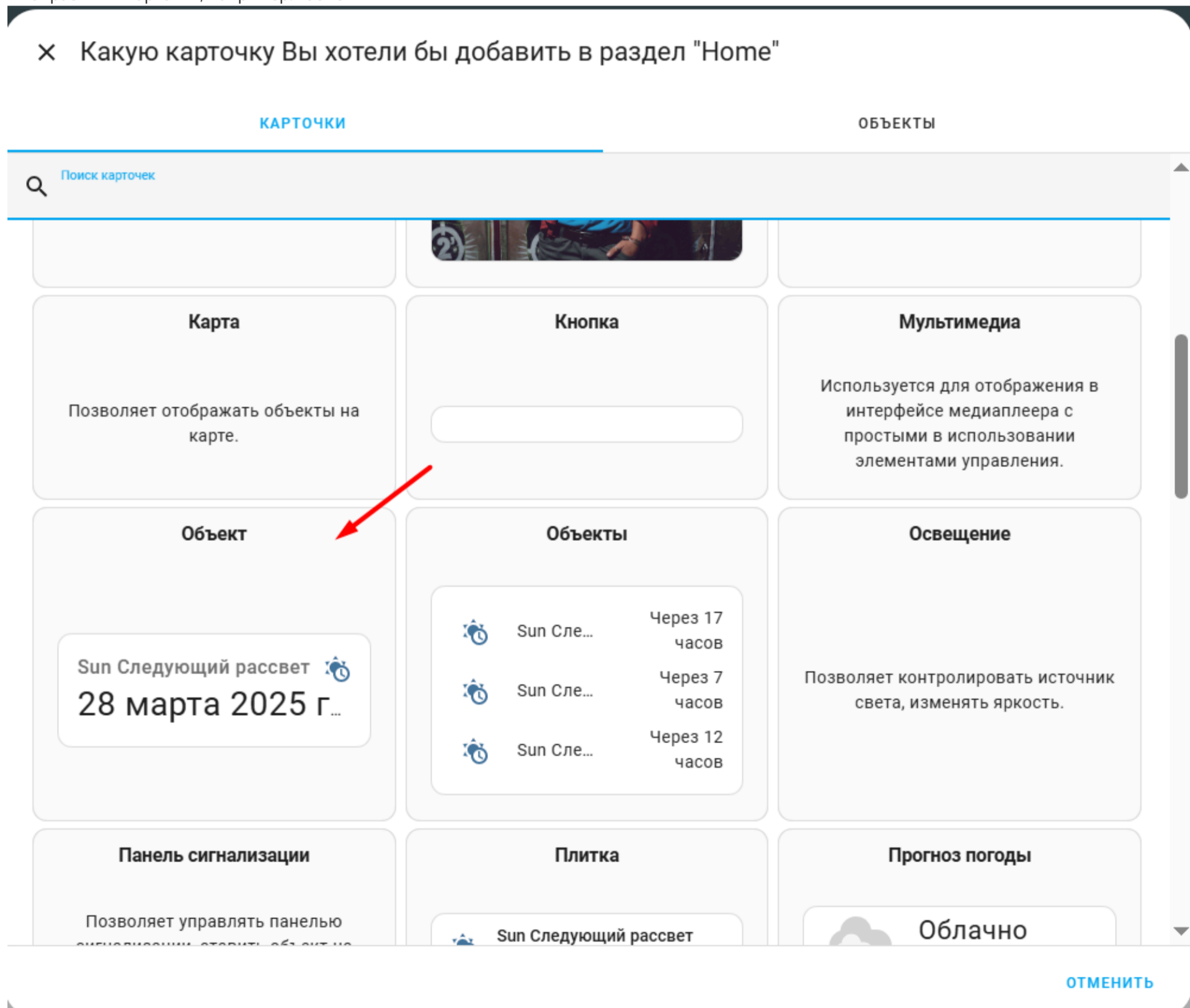
▼ Шаг 2

Добавляем новый дашборд/карточку



Шаг 3

Выбираем тип карточки, например: "объект"



Шаг 4

✕ Настройка карточки "Объект" ?

НАСТРОЙКИВИДИМОСТЬ

Освещенностьlux👁

Объект* Um_lux ✕ ▾	
Название Освещенность	👁 Иконка ▾
Атрибут Unit of measurement ✕ ▾	Единица измерения
Тема (необязательно) ▾	Показывать цвет состояния <input type="checkbox"/>

ТЕКСТОВЫЙ РЕДАКТОРОТМЕНИТЬ СОХРАНИТЬ

