|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01**

**Отчет**

**по лабораторной работе № 8**

**Название: Организация клиент-серверного взаимодейсвтия между Golang и PostgreSQL**

**Дисциплина: Языки интернет-программирования**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-31Б |  |  | Н.Е. Мамаев |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | В.Д. Шульман |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2024

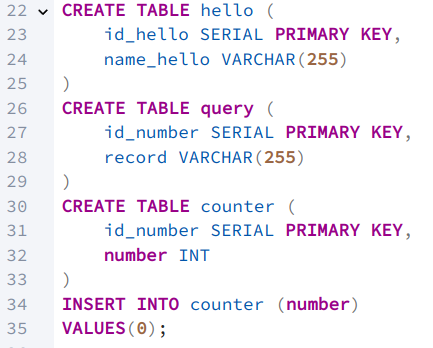
**Цель работы**: получение первичных навыков в организации долгосрочного хранения данных с использованием PostgreSQL и Golang.

**Задание.**

Перекопировать код сервисов, полученный в ходе выполнения 6-й лабораторной работы, в соответствующие поддиректории в директории cmd (кроме кода сервиса hello, т.к. он уже реализован в качестве примера). Доработать сервисы таким образом, чтобы они использовали для хранения данных СУБД PostgreSQL. Каждый сервис должен как добавлять новые данные в БД (insert/update), так и доставать их для предоставления пользователю (select).

1. Создаем БД lw8\_web и в не 3 таблицы.

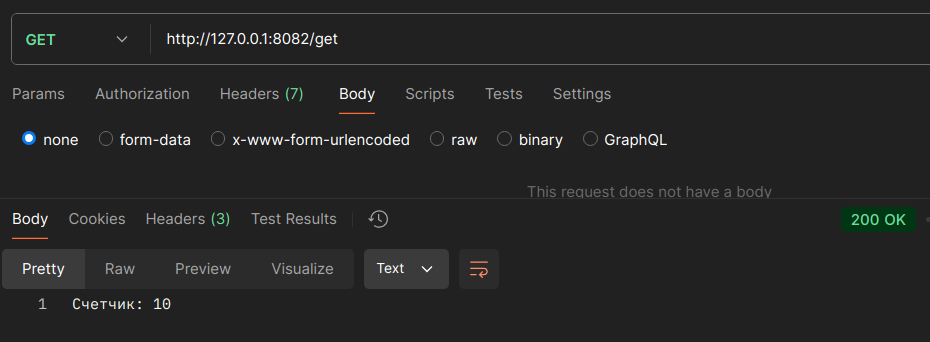
На рисунке 1 изображен скрипт для создания трех таблиц:

Рисунок 1. Создание таблиц.

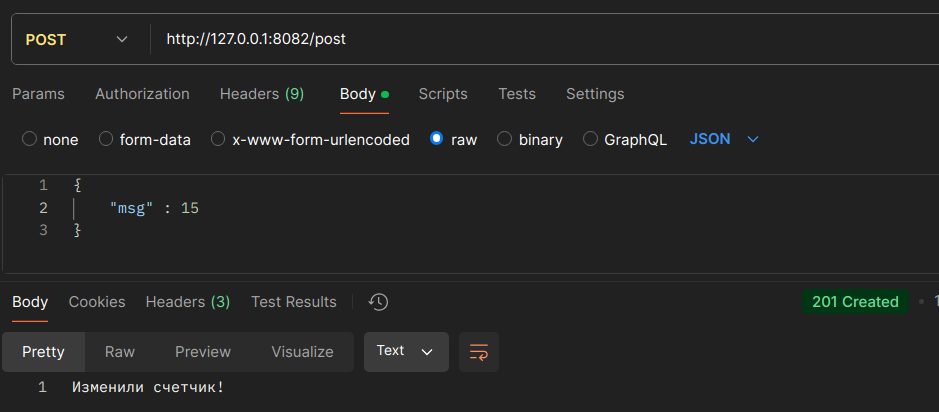
2. Работа микроскервиса Counter.

Примеры работы с микросервисом через Postman.

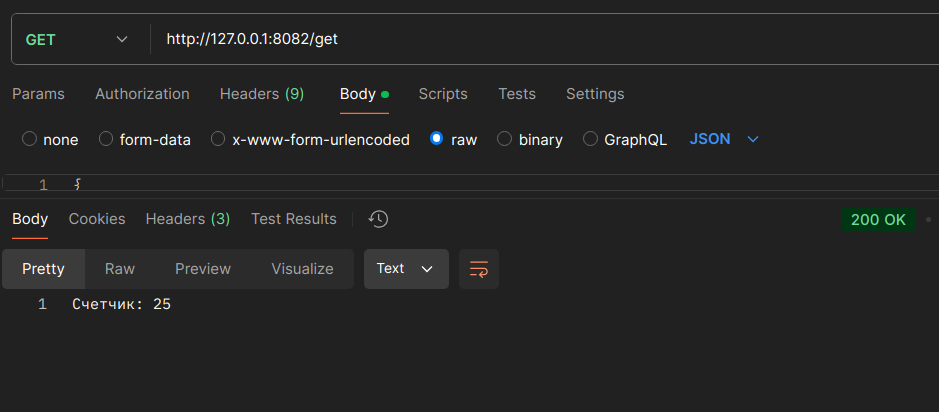
На рисунке 2 изображен результат возврата счетчика, т.е. Get-запроса:

Рисунок 2. Get-запрос, вывод счетчика.

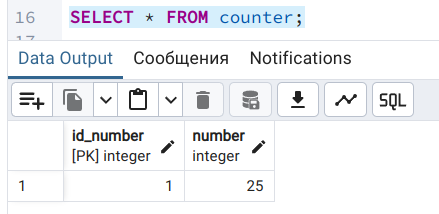
На рисунке 3 изображен результат изменения счетчика, т. е. Post-запроса:

Рисунок 3. Post-запрос, изменение счетчика.

На рисунке 4 изображен результат возврата счетчика, т. е. Get-запроса после изменения:

Рисунок 4. Get-запрос, демонстрация изменения счетчика.

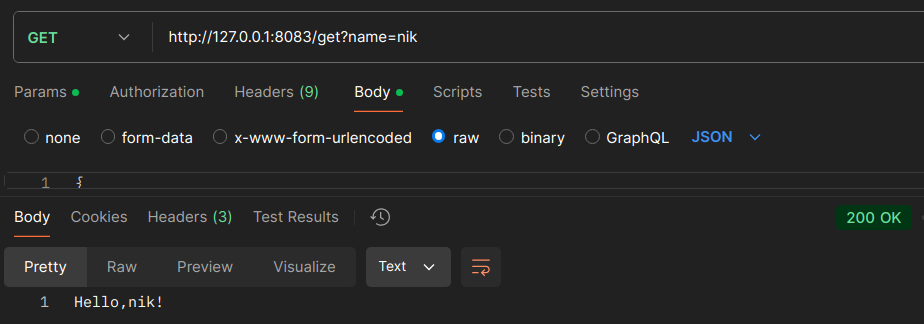
На рисунке 5 продемонстрирована таблица counter:

Рисунок 5. Таблица counter.

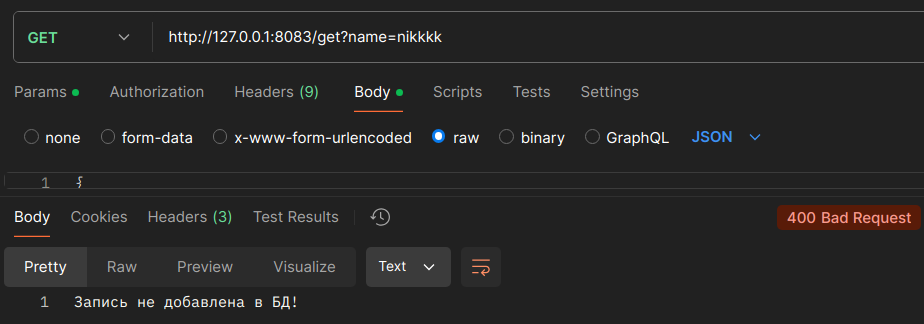
3. Работа микросервиса Query.

Примеры работы с микросервисом через Postman.

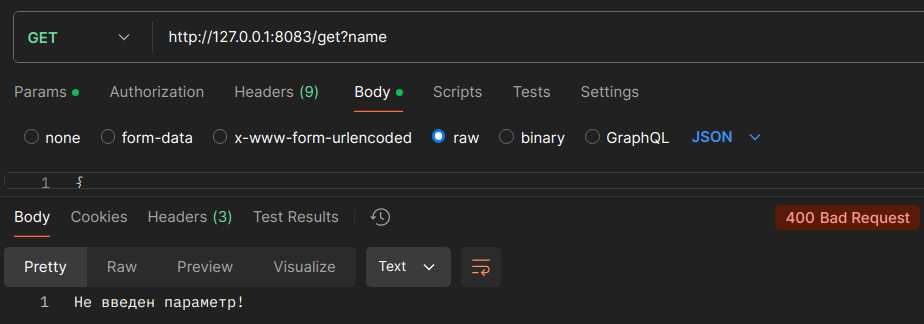
На рисунке 6 изображен результат get-запроса со значением параметра name, которое содержится в таблице:

Рисунок 6. Get-запрос с существующими данными.

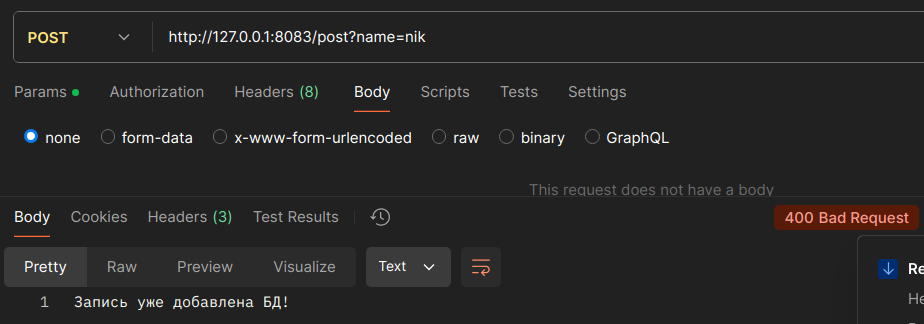
На рисунке 7 изображен результат get-запроса со значением параметра name, который не содержится в таблице:

Рисунок 7. Get-запрос с несуществующими данными.

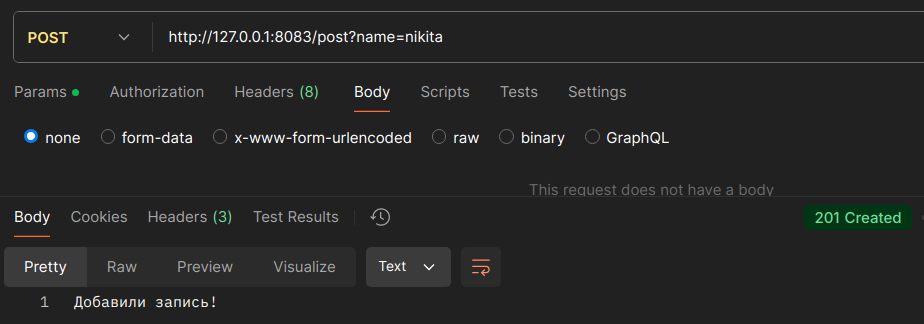
На рисунке 8 изображен результат get-запроса без значения параметра name:

Рисунок 8. Get-запрос без данных.

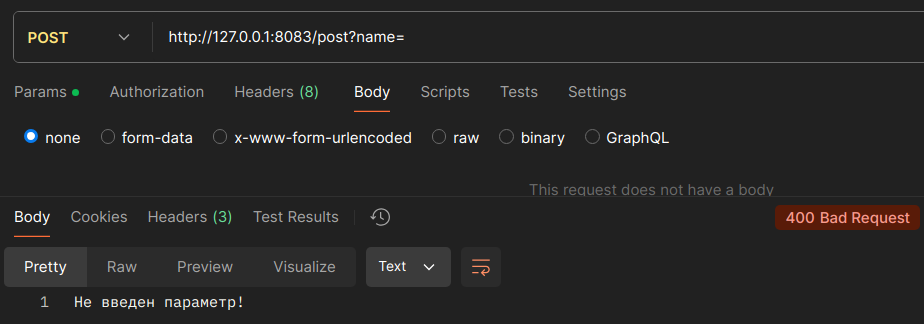
На рисунке 9 изображен результат post-запроса со значением параметра name, который содержится в таблице:

Рисунок 9. Post-запрос с существующими данными.

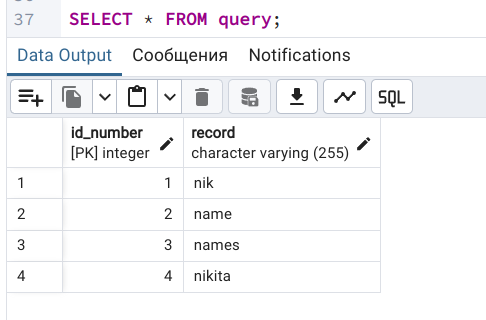
На рисунке 10 изображен результат post-запроса со значением параметра name, который не содержится в таблице:

Рисунок 10. Post-запрос с несуществующими данными.

На рисунке 11 изображен результат post-запроса без значения параметра name:

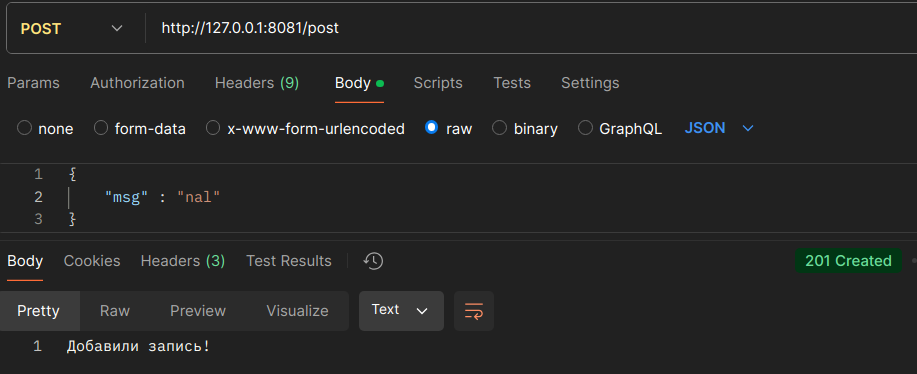
Рисунок 11. Post-запрос без данных.

На рисунке 12 продемонстрирована таблица query:

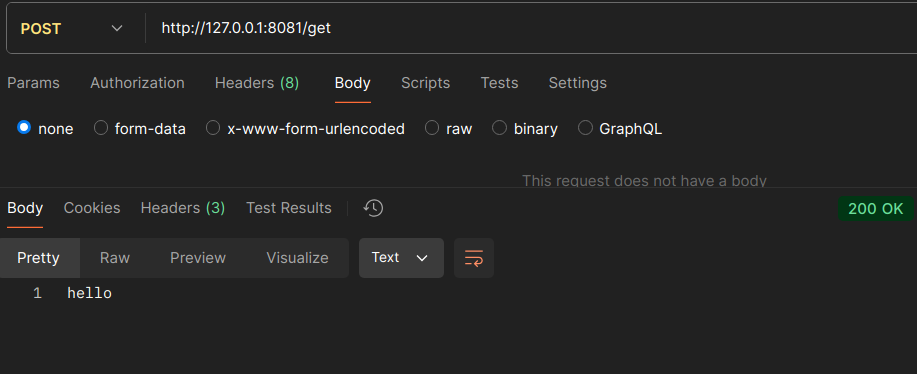
Рисунок 12. Таблица query.

4. Работа микросервиса hello.

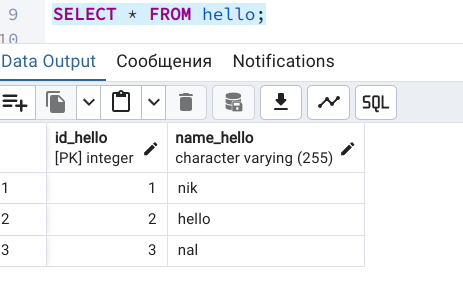
На рисунке 13 изображен результат Post-запроса:

Рисунок 13. Post-запрос, добавление записи.

На рисунке 14 изображен резульат Get-запроса:

Рисунок 14. Get-запрос.

На рисунке 15 изображена таблица hello:

Рисунок 15. Таблица hello.

**Вывод:** в ходе лабораторной работы получены навыки в орагнизации долгосрочного хранения данных с использованием PostgreSQL и Golang.