**Веб-API приложения для размещения медицинский статей**

Приложение позволяет размещать, просматривать, изменять и удалять статьи; размещать, просматривать, изменять и удалять комментарии; получать статистику по количеству комментариев для каждой статьи.

Архитектура приложения:

* Eureka Server – контейнер для публикации микросервисов.
* ApiGateway – шлюз, обрабатывающий входящие запросы и перенаправляющий их на нужный микросервис.
* ArticleService – микросервис, отвечающий за работу со статьями
* CommentaryService – микросервис, отвечающий за работу с комментариями.
* StatisticsService – микросервис, отвечающий за сбор статистики пос статьям.

**ArticleService** *(localhost:8082/article-service)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Название метода** | **Метод** | **End-point** | **Описание** |
| getAllArticles | GET | /articles | Возвращает список всех статей |
| getArticle | GET | /articles/{articleId} | Возвращает статью с указанным ID |
| addArticle | POST | /articles | Добавляет новую статью и возвращает ее URL в заголовке Location |
| updateArticle | PUT | /articles/{articleId} | Изменяет статью с указанным ID |
| deleteArticle | DELETE | /articles/{articleId} | Удаляет статью с указанным ID |

**CommentaryService** *(localhost:8082/commentary-service)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Название метода** | **Метод** | **End-point** | **Описание** |
| get Commentary | GET | /articles/{articleId} /commentaries /{commentaryID} | Возвращает комментарий с указанным ID |
| getArticle Commentaries | GET | /articles/{articleId} /commentaries | Возвращает комментарии к статье с указанным ID |
| add Commentary | POST | /articles/{articleId} /commentaries | Добавляет комментарий к статье с указанным ID и возвращает его URL в заголовке Location |
| update Commentary | PUT | /articles/{articleId} /commentaries /{commentaryID} | Изменяет комментарий с указанным ID |
| delete Commentary | DELETE | /articles/{articleId} /commentaries /{commentaryID} | Удаляет комментарий с указанным ID |

**CommentaryService** *(localhost:8082/statistics-service)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Название метода** | **Метод** | **End-point** | **Описание** |
| get Statistics | GET | /statistics | Возвращает Map<String,Integer> с названием статей и количеством комментариев под ними |

**База данных**

Для хранения информации была выбрана СУБД PostgreSQL.

Название “medicine-db”.

Скрипт создания:

CREATE TABLE "article" (

"id" bigserial PRIMARY KEY,

"title" varchar(100) UNIQUE,

"short\_description" varchar(300),

"article\_text" text,

"author" varchar(100),

"publishing\_date" date

);

CREATE TABLE "commentary" (

"id" bigserial PRIMARY KEY,

"article\_id" bigint,

"author" varchar(100),

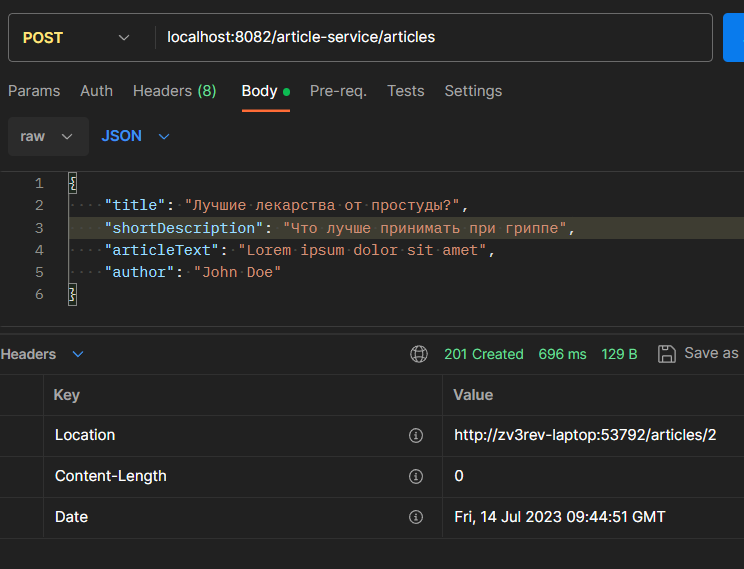
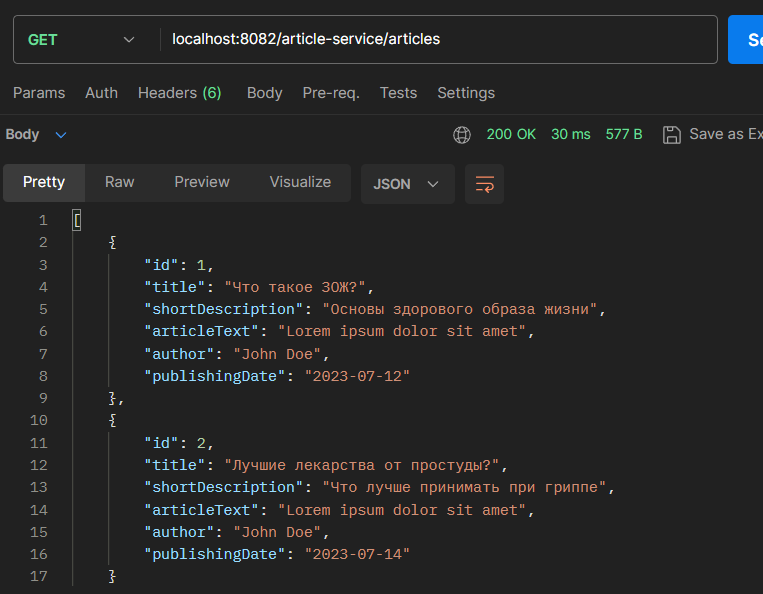
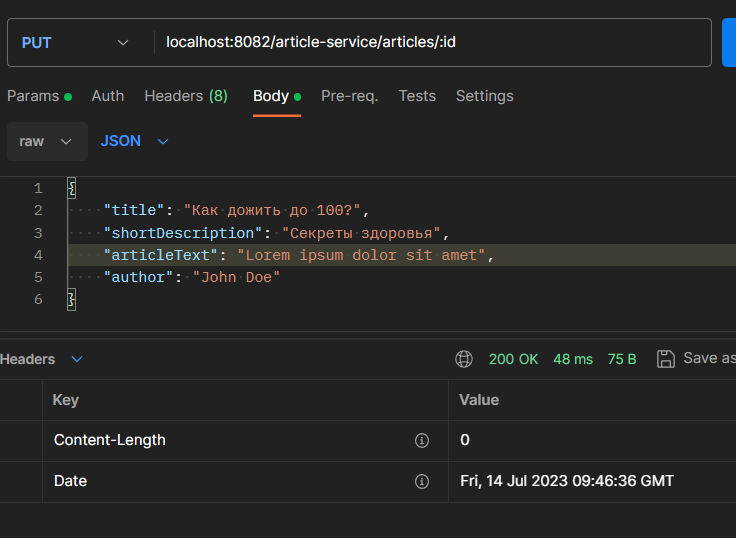
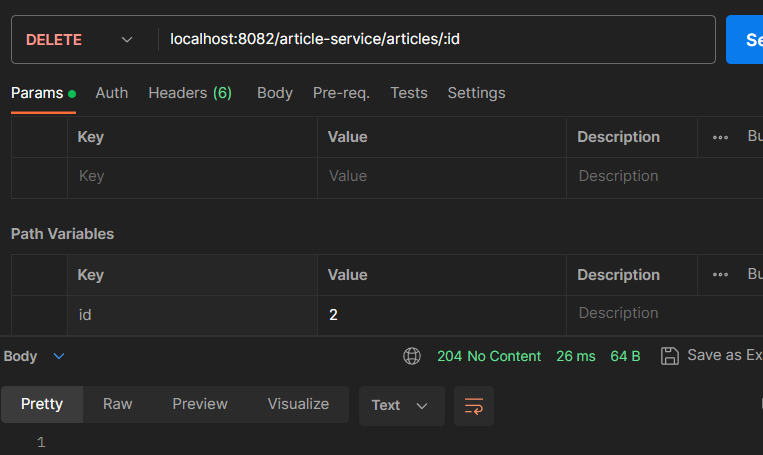
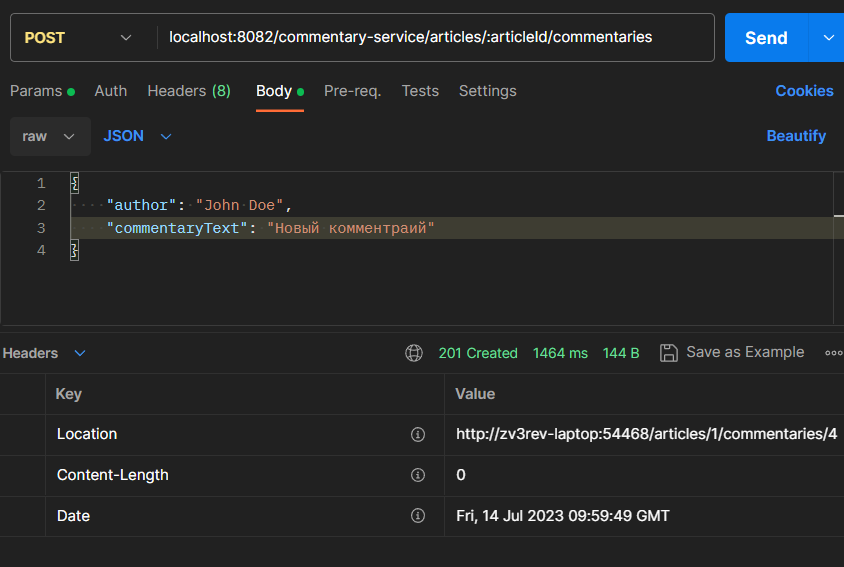
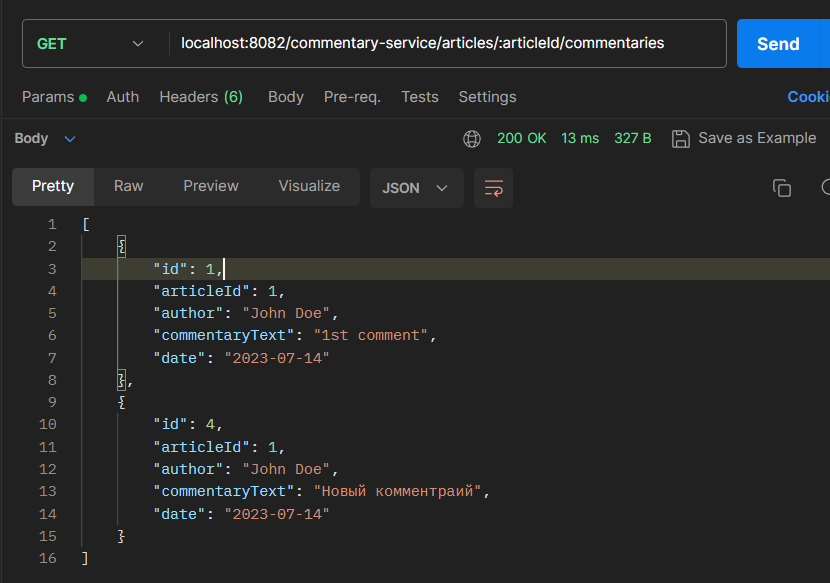
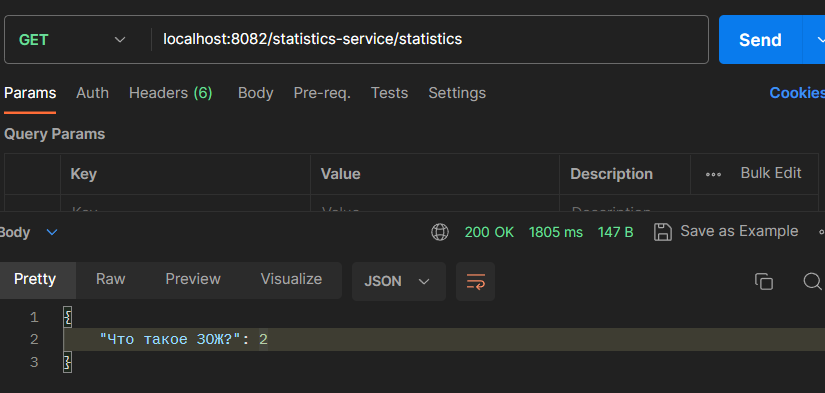
"commentary\_text" text,

"publishing\_date" date

);

ALTER TABLE "commentary" ADD FOREIGN KEY ("article\_id") REFERENCES "article" ("id");

**Пример работы веб-API**

1. Добавление статьи  
   
2. Получение списка статей  
   
3. Изменение статьи  
   
4. Удаление статьи  
   
5. Добавление комментария  
   
6. Просмотр комментариев  
   
7. Получение статистики  
   

P.S. Это первое приложение с микросервисной архитектурой которое я создал и с этим связаны некоторые проблемы в проектах микросервисов. Так же у меня не получилось настроить и подключится к базе H2, поэтому был вынужден был выбрать ту, с которой уже умею работать. Так же не получилось разобраться в создании .csv файлов и отправке их по HTTP.