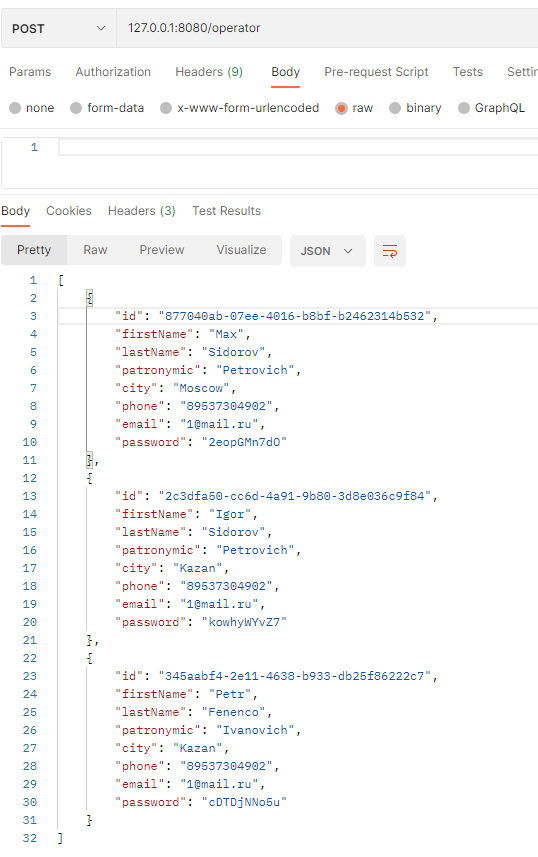
Список всех http-методов:

Сущность «Оператор»

**GET**

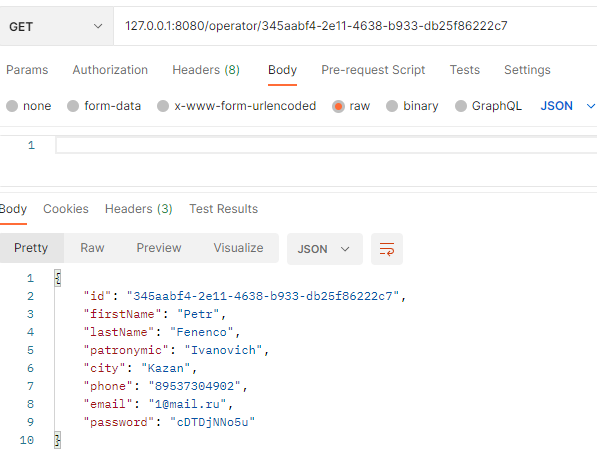
1)url: /operator

Используется для получения информации из базы данных (PostgreSQL). Выводится вся информация об операторах из таблицы «operators».

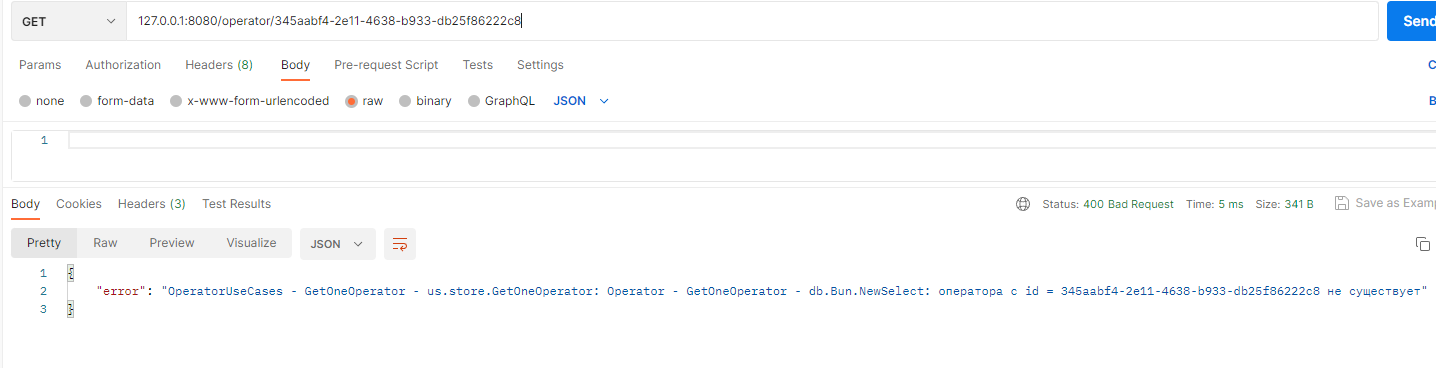


2) url: /operator/:id

Используется для получения информации из базы данных (PostgreSQL). Передавая значение после «/», будет выведена информация об операторе из таблицы «operators» с конкретным operator\_id\_pk, если таковой имеется.



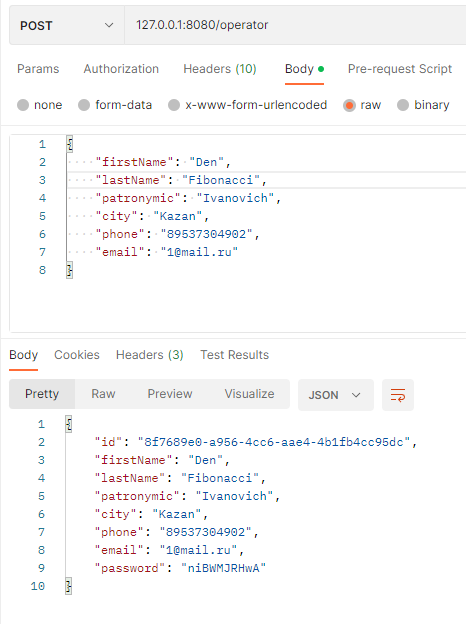
В случае если оператор с таким id не существует в базе:



**POST**

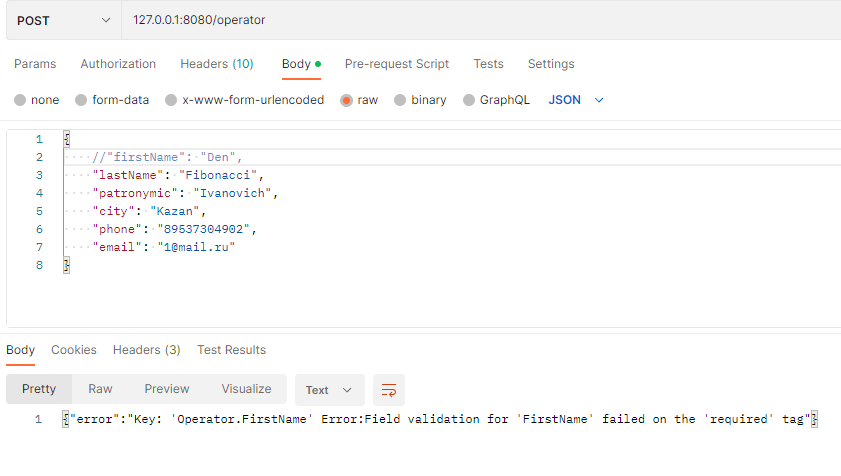
3)url: /operator

Используется для отправки информации в базу данных. Содержит тело, которое может быть в виде xml / html / json или в простом текстовом формате. В нашем случае используется формат json для передачи данных.

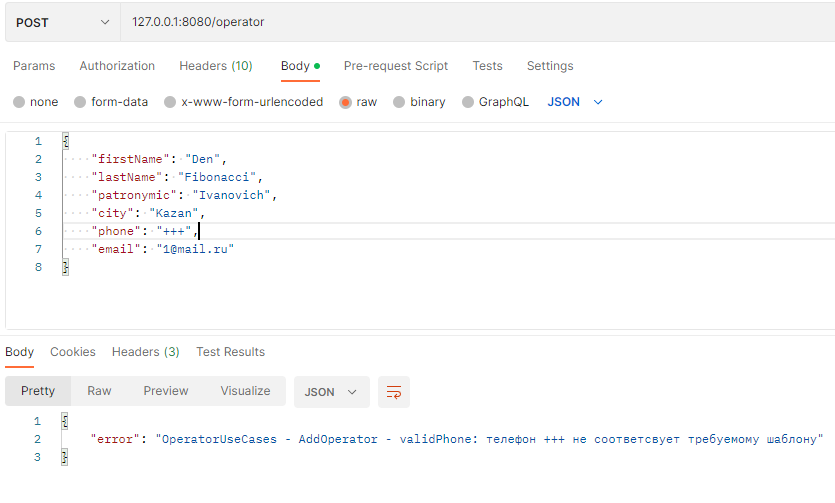


Если информация успешно загружена в базу, получаем в ответ созданную запись со сгенерированным id и паролем.

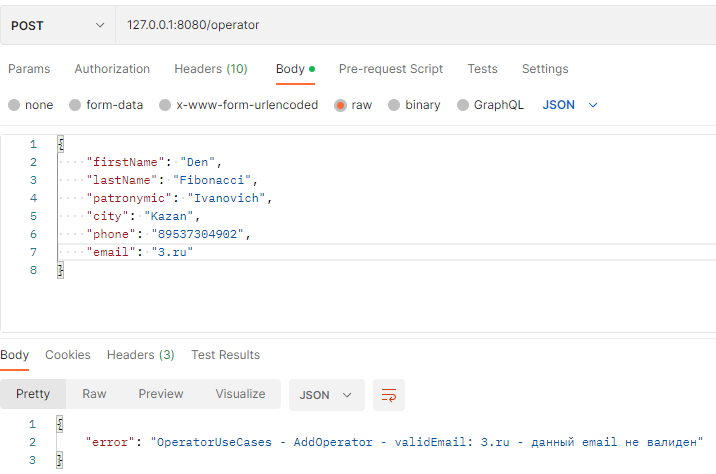
В случае отправки не валидных данных:



- Если не заполнено хотя бы одно поле



- Если не валидно передано значение номера телефона

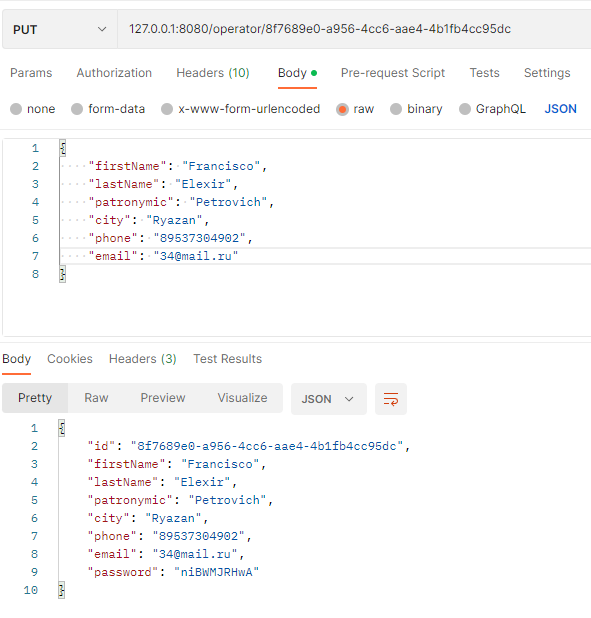


- Если не валидно передано значение email

**PUT**

4)url: /operator/:id

Предназначен для внесения изменений каких-либо данных в базе. Необходимо передать значение id после «/», таким образом будет изменена конкретная запись с передаваемым id.



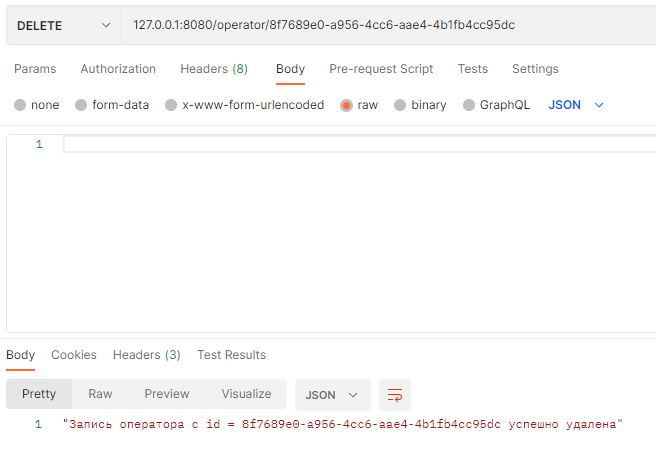
Если запись с указанным id успешно отредактирована, получаем полную запись.

Проверки на валидность загружаемых данных те же, что и у метода **POST**.

**DELETE**

5)url: /operator/:id

Метод позволяет отправить запрос на удаление указанной записи в базе с указанным id после «/».



Если записи нет:

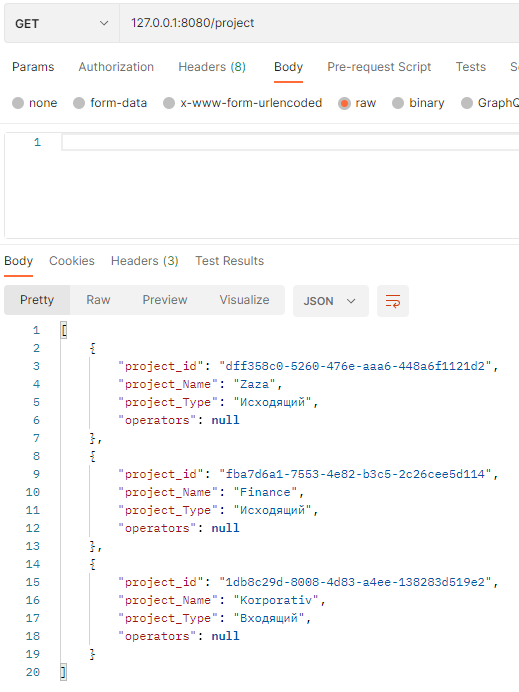


Сущность «Проекты»

**GET**

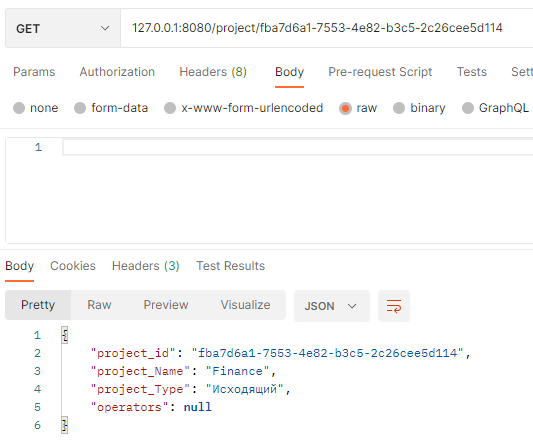
6)url: /project

Используется для получения информации из базы данных (PostgreSQL). Выводится вся информация о проектах из таблицы «projects».

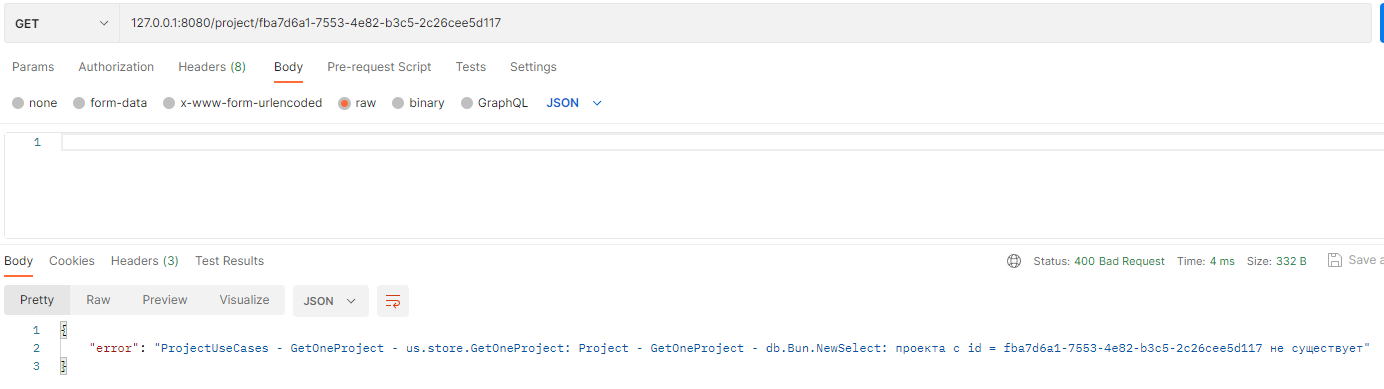


7) url: /project/:id

Используется для получения информации из базы данных (PostgreSQL). Передавая значение после «/», будет выведена информация о проекте из таблицы «projects» с конкретным project\_id\_pk, если таковой имеется.



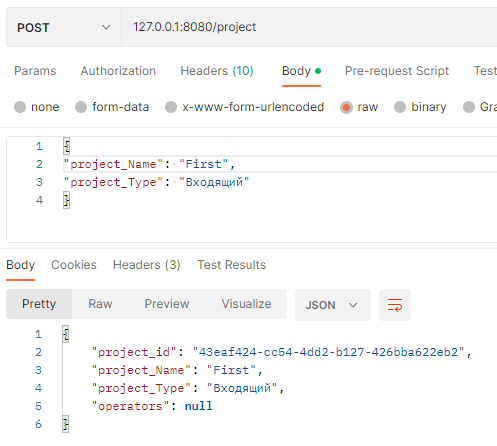
В случае если проекта с таким id не существует в базе:



**POST**

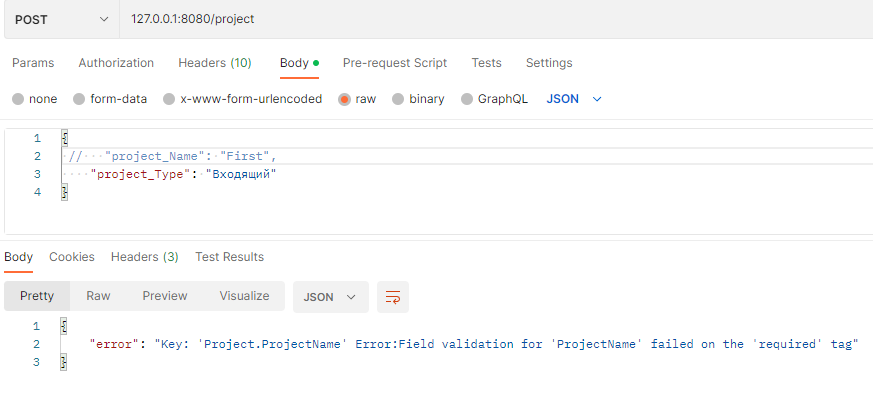
8)url: /project

Используется для отправки информации в базу данных. Содержит тело, которое может быть в виде xml / html / json или в простом текстовом формате. В нашем случае используется формат json для передачи данных.

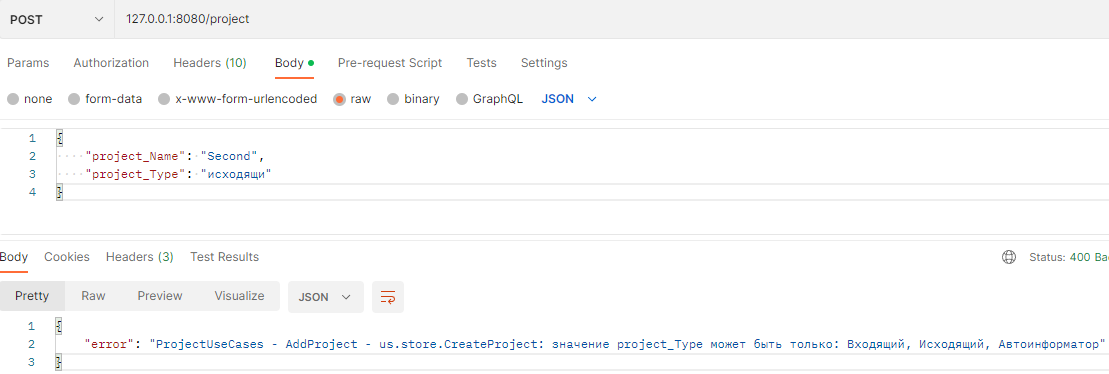


Если информация успешно загружена в базу, получаем в ответ созданную запись со сгенерированным id.

В случае отправки не валидных данных:



- Если не заполнено хотя бы одно поле

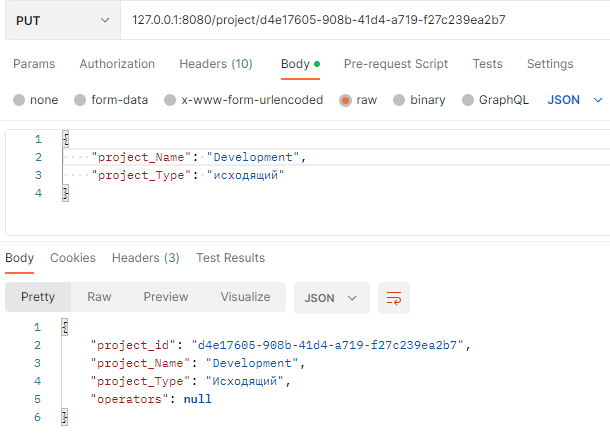


- Если не валидно передано значение project\_type\_fk. Значение project\_type\_fk может быть только: входящий/исходящий/автоинформатор

**PUT**

9)url: /project/:id

Предназначен для внесения изменений каких-либо данных в базе. Необходимо передать значение id после «/», таким образом будет изменена конкретная запись с передаваемым id.



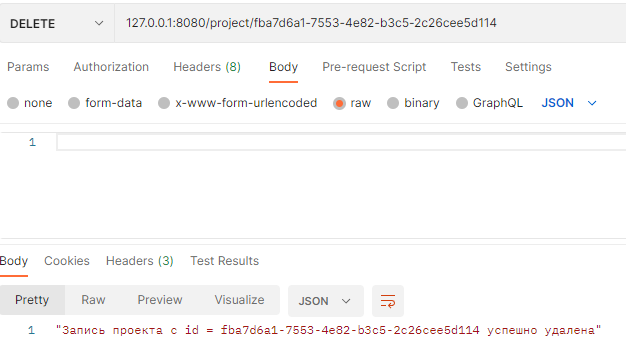
Если запись с указанным id успешно отредактирована, получаем полную запись.

Проверки на валидность загружаемых данных те же, что и у метода **POST**.

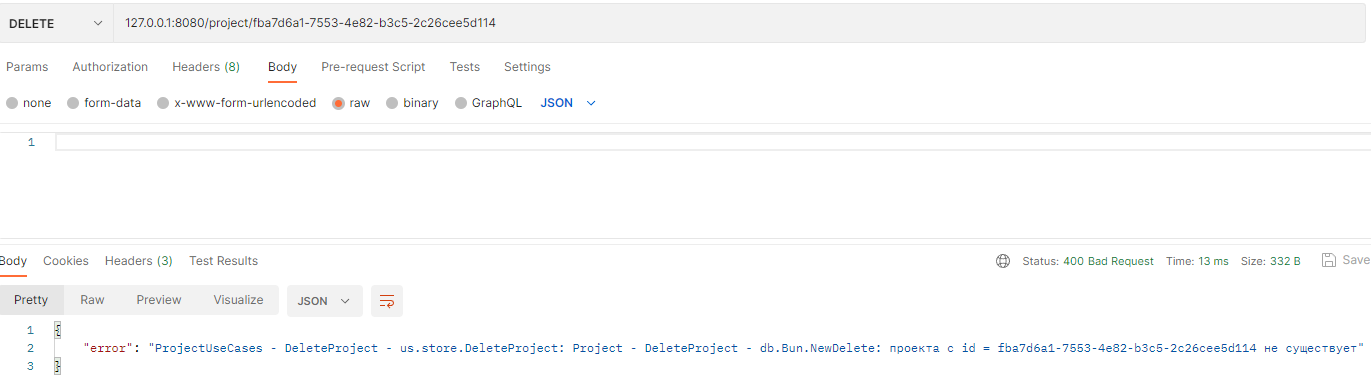
**DELETE**

10)url: /project/:id

Метод позволяет отправить запрос на удаление указанной записи в базе с указанным id после «/».



Если записи нет:



Метод **PUT**

11)url: /AddOperatorToProject/:project\_id/:operator\_id

Запрос используется для добавления операторов в проект.

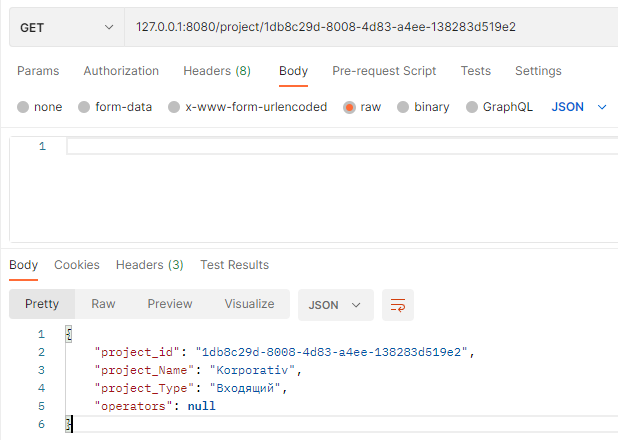
Данный http запрос принимает 2 параметра:

1) project\_id – id проекта

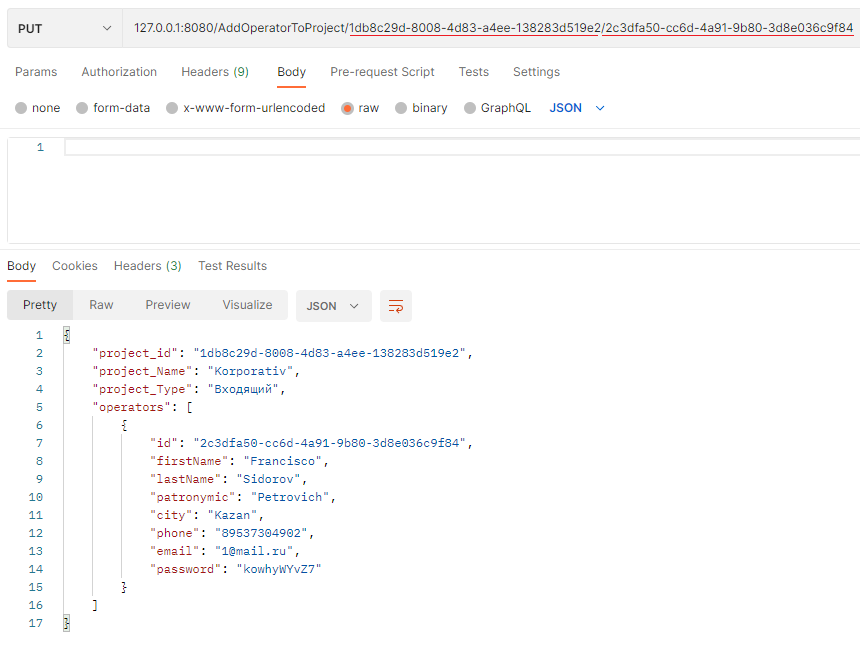
2) operator\_id - id оператора

Метод возвращает отредактированный проект.

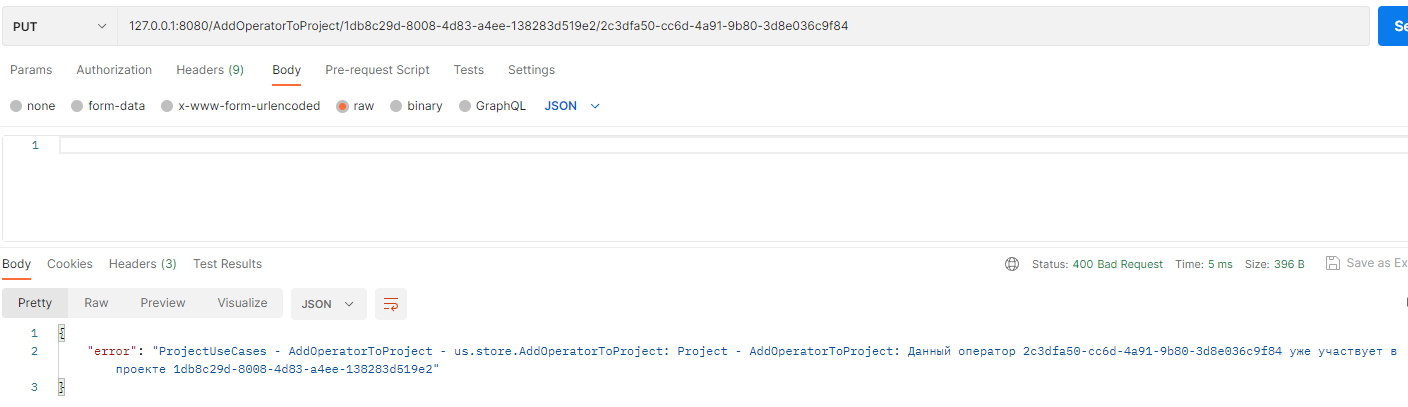
Допустим, есть следующий проект:



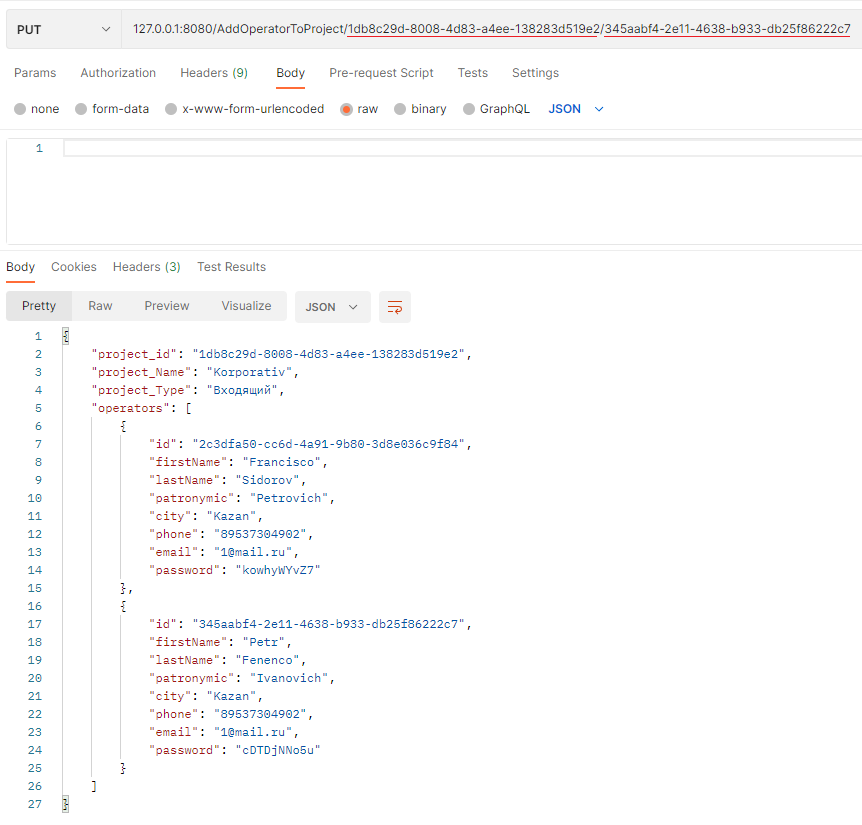
Для добавления оператора в проект необходимо после названия метода через «/» сначала указать id проекта, затем id оператора.



Если оператор уже участвует в проекте:



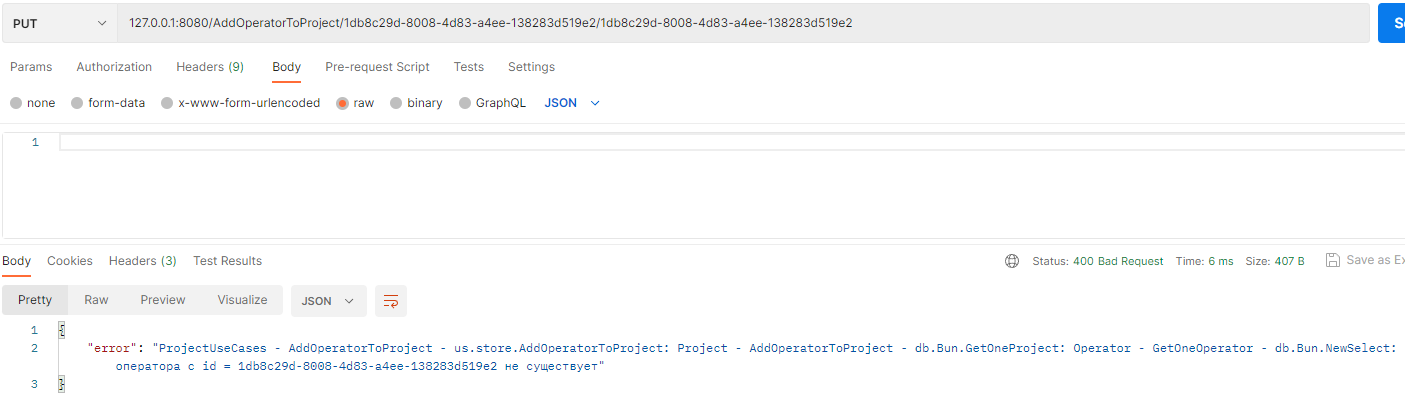
Добавим следующего:



Если проекта, в который мы хотим добавить оператора не существует:



Или же оператора, которого мы хотим добавить в проект не существует:



12) url: /DelOperatorFromProject/:project\_id/:operator\_id

Запрос используется для удаления операторов из проекта.

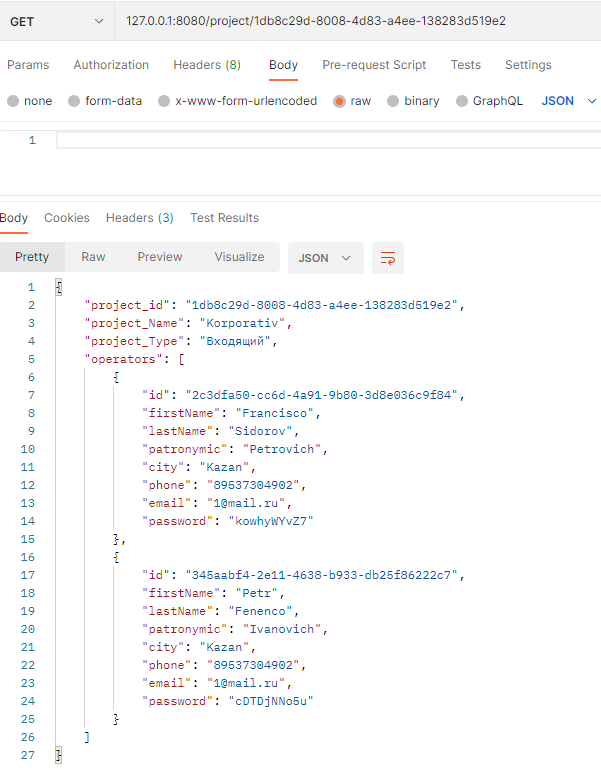
Данный http запрос принимает 2 параметра:

1) project\_id – id проекта

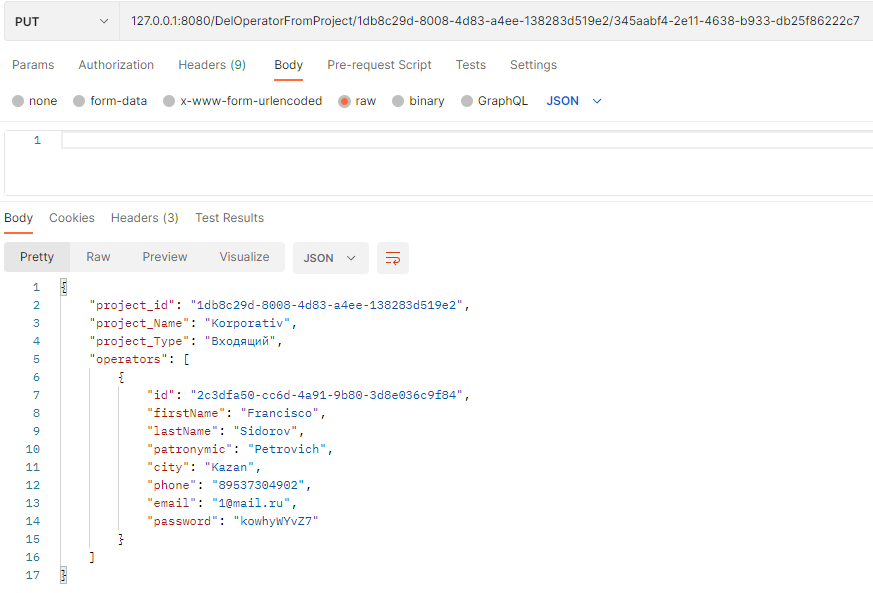
2) operator\_id - id оператора

Метод, как и предыдущий, возвращает отредактированный проект.

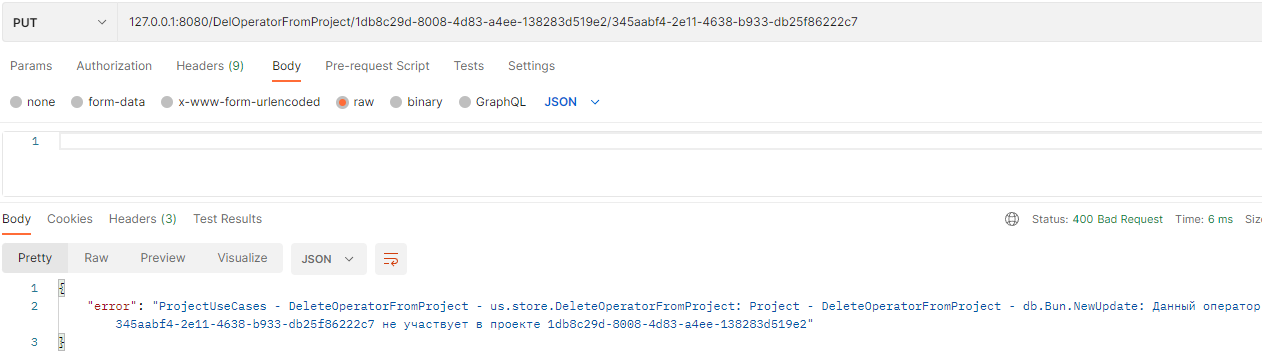
Вернёмся к уже созданному и отредактированному проекту:



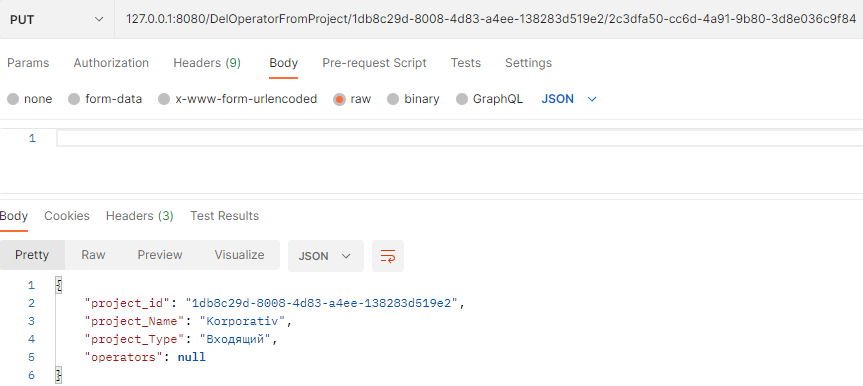
Для удаления оператора из проекта необходимо после названия метода через «/» сначала указать id проекта, затем id оператора.



Если оператор не участвует в проекте:



Удалим ещё одного:



Если не существует проекта из которого мы хотим удалить оператора:

