**Документация с описанием тестовых rest-сервисов**

1. Запросы между собой никак не взаимосвязаны. Если в одном вы удаляете или добавляете пользователя, это не значит, что на запрос получения пользователя вернется измененная информация
2. Host - my-test-pr.herokuapp.com
3. Port – отсутствует, тк является стандартным
4. Страничка swagger - http://my-test-pr.herokuapp.com/swagger-ui.html
5. **Get-запросы**
   1. **Получение списка пользователей**

Описание: возвращает информацию по пользователям

Get-запрос вида http://host:port[/get/users](http://my-test-pr.herokuapp.com/get/users)

Входные параметры: отсутствуют

Выходные параметры: Json-объект с массивом данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | + |
| name | Имя | String | + |
| age | Возраст | Integer | + |

Пример запроса:

**Request**

Get <http://host:port/get/users>

**Response**

Status 200 OK

[

    {

        "id": 1,

        "name": "Oleg",

        "age": 28

    },

    {

        "id": 2,

        "name": "Olga",

        "age": 25

    },

    {

        "id": 3,

        "name": "Kirill",

        "age": 15

    },

    {

        "id": 4,

        "name": "Vika",

        "age": 40

    }

]

* 1. **Получение списка организаций**

Описание: возвращает информацию по организациям и их сотрудникам

Get-запрос вида http://host:port/get/organisations?id={id}

Входные параметры

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | -  Если передан, вернется информация по конкретной организации, иначе по всем организациям. Если организации нет, вернет пустой список |

Выходные параметры: Json-объект с массивом данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | + |
| name | Имя | String | + |
| peopleCount | Кол-во людей в организации | Integer | + |
| peoples | Информация по людям (Структура идентичка users из прошлого запроса) | Json array () | - |

Пример запроса:

**1. Если id не передан**

**Request**

Get <http://host:port/get/organisations>

**Response**

Status 200 OK

[

    {

        "id": 1,

        "name": "Улмарт",

        "peopleCount": 2,

        "peoples": [

            {

                "id": 1,

                "name": "Oleg",

                "age": 28

            },

            {

                "id": 2,

                "name": "Olga",

                "age": 25

            }

        ]

    },

    {

        "id": 2,

        "name": "Алмарт",

        "peopleCount": 1,

        "peoples": [

            {

                "id": 3,

                "name": "Kirill",

                "age": 15

            }

        ]

    },

    {

        "id": 3,

        "name": "Олморт",

        "peopleCount": 0,

        "peoples": **null**

    }

]

**2.** **Если id передан и существует**

**Request**

Get <http://host:port/get/organisations>?id=2

**Response**

Status 200 OK

[

    {

        "id": 2,

        "name": "Алмарт",

        "peopleCount": 1,

        "peoples": [

            {

                "id": 3,

                "name": "Kirill",

                "age": 15

            }

        ]

    }

]

**3.** **Если id передан, но не существует**

**Request**

Get <http://host:port/get/organisations>?id=6

**Response**

Status 200 OK

[]

1. **Post-запросы**
   1. **Создание пользователя**

Описание: Создает пользователя

Post-запрос вида <http://host:port/post/create>

Входные параметры

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | - |
| name | Имя | String | - |
| age | Возраст | Integer | - |

Выходные параметры: Json-объект с массивом данных такого же типа

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | - |
| name | Имя | String | - |
| age | Возраст | Integer | - |

Пример запроса:

**Request**

Post <http://host:port/post/create>

    {

        "id": 10,

        "name": "Vlad"

    }

**Response**

Status 200 OK

{

    "id": 10,

    "name": "Vlad",

    "age": **null**

}

* 1. **Создание множества пользователей**

Описание: Создает нескольких пользователей

Post-запрос вида <http://host:port/post/create>/list

Входные параметры: Json array c параметрами

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | - |
| name | Имя | String | - |
| age | Возраст | Integer | - |

Выходные параметры: Json-объект с массивом данных такого же типа

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | - |
| name | Имя | String | - |
| age | Возраст | Integer | - |

Пример запроса:

**Request**

Post <http://host:port/post/create>/list

[

    {

       "id": 10,

       "name": "Vlad"

    },

    {

       "id": 12,

       "age": 55

    }

]

**Response**

Status 200 OK

[

    {

        "id": 10,

        "name": "Vlad",

        "age": **null**

    },

    {

        "id": 12,

        "name": **null**,

        "age": 55

    }

]

1. **Form-data Post-запрос**
   1. **Отправка вложения**

Описание: Отправляет вложение (прикрепленный файл) на сервер

Post-запрос вида <http://host:port/form/data/file>

Входные параметры: form-data file

Выходные параметры

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| name | Имя | String | + |
| size | Размер | Integer | + |

Пример запроса:

**Request**

Post <http://host:port/form/data/file>

Form-data прикрепленный файл

**Response**

Status 200 OK

{

    "name": "Вопросы.docx",

    "size": 15085

}

1. **Put-запросы**
   1. **Изменение информации о пользователе**

Описание: Изменяет информацию о пользователе

Put-запрос вида <http://host:port/put/user>

Входные параметры

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | +, если передан неверный или не передан, то вернется 400 ошибка |
| name | Имя | String | - |
| age | Возраст | Integer | - |

Выходные параметры: Json-объект с массивом данных такого же типа

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | - |
| name | Имя | String | - |
| age | Возраст | Integer | - |

Пример запроса:

1. **Если передан верный id**

**Request**

Put <http://host:port/put/user>

 {

       "id": 1,

       "name": "Vlad",

       "age": 1

 }

**Response**

Status 200 OK

{

    "id": 1,

    "name": "Vlad",

    "age": 1

}

1. **Если передан не верный id**

Put <http://host:port/put/user>

 {

       "id": 10,

       "name": "Vlad",

       "age": 1

 }

**Response**

Status 400м

{

    "code": 400,

    "description": "id 10 is not found"

}

1. **Delete-запросы**
   1. **Удаление пользователя**

Описание: удаляет пользователя из системы

Delete-запрос вида [http://host:port/delete/user?id={id}](http://host:port/delete/user?id=%7bid%7d)

Входные параметры

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | -  Если передан, из списка пользователей пользователя с этим id и вернет список всех оставшихся пользователей. Если не передан или пользователя с таким id нет, то вернет весь список пользователей |

Выходные параметры: Json-объект с массивом данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| id | Идентификатор | Long | + |
| name | Имя | String | + |
| age | Возраст | Integer | + |

Пример запроса:

**Request**

Delete <http://host:port/delete/user?id=1>

**Response**

Status 200 OK

[

    {

        "id": 2,

        "name": "Olga",

        "age": 25

    },

    {

        "id": 3,

        "name": "Kirill",

        "age": 15

    },

    {

        "id": 4,

        "name": "Vika",

        "age": 40

    }

]

1. **Ошибочные запросы**
   1. **Специальные запросы, которые возвращают ошибку в ответ**

Get-Запросы вида

[http://host:port/errors/301](http://host:port/delete/user?id=1)

<http://host:port/errors/400>

<http://host:port/errors/401>

<http://host:port/errors/500>

Входные параметры: отсутствуют

Выходные параметры: Json-объект с массивом данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Тип | Обязательность |
| code | код | Integer | + |
| description | описани | String | + |