

REST API

Здесь описываются доступные REST API операции. Общий адрес для всех REST API операций: **<http://localhost:9001/api/>**

Группа операций *Intervals*

Обеспечивает доступ к операциям над интервалами. Интервал — это промежуток между днями, когда происходит пополнение бюджета.

Например, если зарплата приходит 10 и 24 числа, то первый интервал будет 10.01.2016 — 24.01.2016, второй интервал: 25.01.2016 — 10.02.2016. и т. д.

JSON объект *interval*

```
{ id: [number], start: [timestamp], end: [timestamp], sum: [decimal] }
```

1. id – идентификатор платежа;
2. start – начало интервала, содержит unix timestamp, например 1454419024160 (соответствует 2 февраля 2016);
3. end – конец интервала;
4. sum – сумма, с которой начинается интервал, например 123.45;

Пример:

```
{ id: 12345, start: 1441129834000, end: 1441138693800, sum:123.45 }
```

GET <http://localhost:9001/api/intervals>

Вернёт последний интервал в списке. **Achtung:** Скоро будет исправлено на возвращение списка интервалов!

GET [http://localhost:9001/api/intervals?timestamp=\[timestamp\]](http://localhost:9001/api/intervals?timestamp=[timestamp])

Запрос что-бы получить интервал, на который приходится указанная метка времени.

Пример: <http://localhost:9001/api/intervals?timestamp=1454419024160>

GET [http://localhost:9001/api/intervals?from=\[date\]&till=\[date\]](http://localhost:9001/api/intervals?from=[date]&till=[date])

Запрос, что-бы получить список интервалов между двумя датами. Пример:

<http://localhost:9001/api/intervals?from=2015-12-01&till=2015-01-31>

GET <http://localhost:9001/api/intervals/latest>

Вернёт последний интервал в списке

GET <http://localhost:9001/api/intervals/latest/summary>

Вернёт последний интервал, список платежей и статистику.

GET <http://localhost:9001/api/intervals/:id>

Получить интервал по заданному **id**. Пример запроса:
<http://localhost:9001/api/intervals/12345>

GET <http://localhost:9001/api/intervals/:id/payments>

Вернёт список платежей, которые принадлежат интервалу, выбранному по **id**. Пример запроса: <http://localhost:9001/api/intervals/12345/payments>

POST <http://localhost:9001/api/intervals/>

Добавить в список интервалов новый объект типа *Interval*.

DELETE <http://localhost:9001/api/intervals/:id>

Удалить из списка интервалов интервал с заданным **id**.

Группа операций Payments

Обеспечивает доступ к операциями с платежами.

JSON объект payment

```
{ id: [number], date: [text], time: [number], sum: [number] }
```

1. id – идентификатор платежа;
2. date – дата платежа в формате **yyyy-mm-dd**;
3. time – временная метка платежа в формате unix timestamp;
4. sum – сумма платежа.

Пример:

```
{ id: 12345, date: "2015-12-31", time: 1441129834000, sum: 12.45 }
```

GET <http://localhost:9001/api/payments>

Получить список расходов из последнего интервала;

GET [http://localhost:9001/api/payments?id=\[number\]](http://localhost:9001/api/payments?id=[number])

Получить платёж по **id**.

GET [http://localhost:9001/api/payments?date=\[date\]](http://localhost:9001/api/payments?date=[date])

Получить список платежей по дате. Дата задаётся в формате **yyyy-mm-dd**. Пример:
<http://localhost:9001/api/payments?date=2015-01-31>

GET [http://localhost:9001/api/payments?from=\[date\]&till=\[date\]](http://localhost:9001/api/payments?from=[date]&till=[date])

Получить список платежей совершённых между датами, заданными from и till. Дата заданна в формате **yyyy-mm-dd**.

POST <http://localhost:9001/api/payments>

Сохранить объект или список объектов типа payment в списке платежей.

DELETE <http://localhost:9001/api/payments/:id>

Удалить платёж по его **id**.