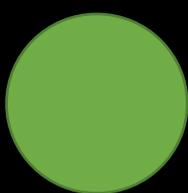


Яндекс Заправки



Программный интерфейс корпоративного клиента

Справочник разработчика

Версия 1.2

<https://tanker.yandex.ru>

dev@tanker.yandex.ru

ООО Яндекс.Заправки

Содержание

Протокол для интеграции корпоративного клиента	3
Получение списка сотрудников	3
Получение данных по сотруднику.....	4
Балансы сотрудников	4
Текущий баланс сотрудника	5
Пополнить или уменьшить баланс сотрудника	5
Получение отчета о заказах	6

Протокол для интеграции корпоративного клиента

Для интеграции системы на стороне корпоративного клиента необходимо реализовать следующие функции

Интегрируемая система предоставляет базовый url и API ключ для отправки запросов на сервера системы

При ответе сервера любым статусом, кроме 200 OK, система TANKER пытается 5 раз повторить запрос. Таймаут запроса составляет 10 секунд.

Набор нескольких схожих функций обусловлен оптимизаций трафика

Получение списка сотрудников

Базовый url + /tanker/api/user/list?apikey=<API>

В ответ интегрируемая система дает ответ в формате JSON

```
{
  // идентификатор
  "1017b49de51ce211bb700030487e046b":
  {
    // имя
    "FirstName":"ГЕННАДИЙ",
    // фамилия
    "LastName":"ВЫШЕБАБИН",
    // отчество
    "Surname":"АЛЕКСЕЕВИЧ",
    // статус работы
    "WorkStatusType":"Working",
    // телефон
    "Phones":"+79037334954",
    // дата создания записи
    "DateCreate":"2017-04-21T07:00:26.349Z"
  },
  ..... N
}
```

Параметр WorkStatusType может принимать следующие значения

```
public enum WorkStatusType
{
  // Уволен
  Fired = 0,
  // Временно не работает
  NotWorking = 1,
  // Работает
  Working = 2
}
```

Получение данных по сотруднику

Базовый url + /tanker/api/user/item?apikey=<API>&id=<ID>

В ответ интегрируемая система дает ответ в формате JSON

```
{
    // текущий баланс
    "balance":6129.22,
    // текущий лимит на баланс
    "limit":100.00,
    // имя
    "FirstName":"ГЕННАДИЙ",
    // фамилия
    "LastName":"ВЫШЕБАБИН",
    // отчество
    "Surname":"АЛЕКСЕЕВИЧ",
    // статус работы
    "WorkStatusType":"Working",
    // телефон
    "Phones":"+79037334954",
    // дата создания записи
    "DateCreate":"2017-04-21T07:00:26.349Z"
}
```

limit – описывает несжимаемую часть баланса, он может быть как положительным так и отрицательным

Балансы сотрудников

Базовый url + /tanker/api/user/balance/list?apikey=<API>

В ответ интегрируемая система дает ответ в формате JSON

Dictionary<string, double>

```
{
    "041f371e1afa4e199688efc49455e0ff": 6129.22,
    "0a342187a97f42e09eac94da6aa5a9a6": 327.62,
    "112e008281b94853a3034058953b4379": 586.50,
    .... N
}
```

ключ – идентификатор

значение – текущий баланс

Текущий баланс сотрудника

Базовый url + /tanker/api/user/balance/item?apikey=<API>&id=<ID>

В ответ интегрируемая система дает ответ в формате JSON

```
{
    // текущий баланс
    "balance":6129.22,
    // текущий лимит на баланс
    "limit":100.00
}
```

limit – описывает несжимаемую часть баланса, он может быть как положительным так и отрицательным

Пополнить или уменьшить баланс сотрудника

Пополнить баланс

Базовый url +

/tanker/api/user/balance/plus?apikey=<API>&id=<ID>&id_pay=<ID_PAY>&sum=<SUM>&description=<DESCRIPTION>

Уменьшить баланс

Базовый url + /tanker/api/user/balance/minus?apikey=<API>&id=<ID>&id_pay=<ID_PAY>&sum=<SUM>&description=<DESCRIPTION>

id_pay – идентификатор платежа, генерируется системой TANKER

Ответ

Результат обработки описывается кодом HTTP-ответа.

200 OK — всегда обозначает, что запрос выполнен полностью без ошибок;

400 Bad Request — всегда обозначает, что в запросе не хватает параметров (в URI или теле запроса), значение параметров указано неверно.

401 Duplicate— если платеж с указанным id_pay уже имеется в интегрируемой системе

500 Bad request – внутренняя ошибка сервера интегрируемой системы

Получение отчета о заказах

Для получения отчетов о заказах необходимо использовать следующие базовые адреса

Для developer зоны базовый url <http://app.tst.tanker.yandex.net>

Для production зоны базовый url <https://app.tanker.yandex.net>

Отчет возвращается в формате JSON, массив объектов Order

базовый url + /api/order/list

POST

apikey={apikey}&sdate={sdate}&edate={edate}&page={page}

все параметры являются обязательными

Тело ответа запроса содержит JSON вида

```
// флаг наличия следующего листа отчета
bool IsNextPage,
// кол-во данных на листе
int PageSize,
// номер листа
int PageIndex,
// массив данных
List<Order> Items
```

Тело запроса содержит JSON вида Order

```
{
  // идентификатор заказа
  string Id
  // идентификатор пользователя
  string UserId
  // дата и время создания в Utc
  DateTime DateCreate
  // тип заказа
  OrderType OrderType
  // значение заказа
  double OrderVolume
  // идентификатор станции АЗС
  string StationId,
  // номер колонки
  int ColumnId
  // идентификатор топлива
  string FuelId
  // наименование марки топлива
  string FuelMarka
  // наименование брендированой марки топлива
  string FuelName
  // стоимость 1 литра топлива
  double PriceFuel
  // сумма заказа
  double Sum
```

```
// кол-во литров
double Litre
// сумма оплаченная картой
double SumPaidCard
// итого оплачено
double SumPaid
// статус заказ
OrderStatus Status
// дата и время завершения заказа Utc
DateTime DateEnd
// идентификатор причины отмены заказа
string ReasonId
// причина отмены
string Reason
// итого сумма литров залито
double LitreCompleted
// итого оплачено картой с учетом рефанда по завершению заказа
double SumPaidCardCompleted
// итого оплачено по завершению заказа
double SumPaidCompleted
}
```

sdate и edate – период отчета время по Московскому региону

выборка идет по границам

{order.DateEnd} <= sdate AND {order.DateEnd} < edate

возможные форматы "dd.MM.yyyy HH:mm:ss", "dd.MM.yyyy"

page – номер листа отчета по умолчанию = 0

При формировании отчета за период система максимально отдает 1000 строк данных

В ответе имеется параметр IsNextPage который сообщает о наличие следующего листа в отчете
Отчет необходимо получать циклом до тех пор, пока IsNextPage = true