TimeControl RestAPI (версия 1.7)

Оглавление

[Подключение RestAPI 3](#_Toc161048418)

[Настойка подключения 3](#_Toc161048419)

[Пример вызовы 3](#_Toc161048420)

[Возврат ошибки 4](#_Toc161048421)

[Системные структуры 4](#_Toc161048422)

[Структура поиска сотрудника PersonSearch 4](#_Toc161048423)

[Сервисные функции 4](#_Toc161048424)

[Проверка связи (testconnect) 4](#_Toc161048425)

[Выполнение SQL запроса (select) 5](#_Toc161048426)

[Сотрудники 6](#_Toc161048427)

[Справочник должностей (doljnost) 6](#_Toc161048428)

[Добавить должность (doljnost/add) 7](#_Toc161048429)

[Удалить должность (doljnost/delete) 7](#_Toc161048430)

[Справочник подразделений (department) 8](#_Toc161048431)

[Добавить подразделение (department/add) 9](#_Toc161048432)

[Удалить подразделение (department/delete) 10](#_Toc161048433)

[Справочник сотрудников (persons) 10](#_Toc161048434)

[Добавить сотрудника (persons/add) 12](#_Toc161048435)

[Редактировать сотрудника (persons/edit) 13](#_Toc161048436)

[Увольнение\восстановление сотрудника (persons/edit) 13](#_Toc161048437)

[Удалить сотрудника (persons/delete) 14](#_Toc161048438)

[Получить фото сотрудника (persons/foto/get) 15](#_Toc161048439)

[Загрузить фото сотрудника (persons/foto/set) 15](#_Toc161048440)

[Удалить фото сотрудника (persons/foto/delete) 16](#_Toc161048441)

[Выручка 16](#_Toc161048442)

[Выручка из журнала «Заработная плата - Выручка» (revenue) 16](#_Toc161048443)

[Добавить выручку в журнал (revenue/add) 18](#_Toc161048444)

[Удалить выручку из журнала (revenue/delete) 19](#_Toc161048445)

[События регистрации 20](#_Toc161048446)

[События регистрации за период (fact\_events) 20](#_Toc161048447)

[Добавить событие регистрации (fact\_events/add) 22](#_Toc161048448)

[Учет рабочего времени 23](#_Toc161048449)

[Интервалы присутствия (time/intervals) 23](#_Toc161048450)

[Присутствующие на данный момент сотрудники (time/intervals) 25](#_Toc161048451)

[Получение рабочего времени (план\факт) по дням (time/works) 25](#_Toc161048452)

[Импорт данных в TimeControl (в том числе для обмена с 1С) 28](#_Toc161048453)

[Импорт должностей списком (doljnost/addfromlist) 28](#_Toc161048454)

[Импорт справочника юридических лиц списком (company/addfromlist) 29](#_Toc161048455)

[Импорт подразделений списком (department/addfromlist) 30](#_Toc161048456)

[Импорт сотрудников списком (persons/addfromlist) 31](#_Toc161048457)

[Импорт отсутствий списком (absences/addfromlist) 34](#_Toc161048458)

[Получения списка UID сотрудников из TimeControl для задания фото по списку (persons/needfotolist) 36](#_Toc161048459)

[Загрузка фотографии сотрудника по UID (persons/setfoto) 37](#_Toc161048460)

[Экспорт данных из TimeControl (в том числе для обмена с 1С) 38](#_Toc161048461)

[Экспорт отгулов и прогулов списком (absences/getto1c) 38](#_Toc161048462)

[Экспорт табеля (persons/gettabel) 40](#_Toc161048463)

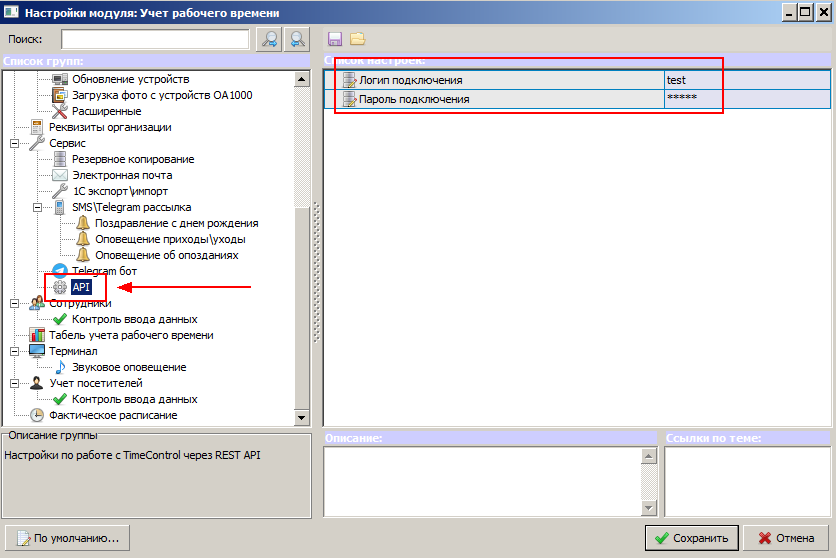
[ИСТОРИЯ ИЗМЕНЕНИЯ ВЕРСИЙ RestAPI 42](#_Toc161048464)

# Подключение RestAPI

## Настойка подключения

RestAPI доступна с версии **TimeControl 5.5**. За работу HTTP сервера RestAPI отвечает служба **«TimeControlWebService».** Для работы на сервере обращение осуществляется по адресу 127.0.0.1 на порт 5053. Для работы по сети используется внешний IP адрес сервера TimeControl.

Для работы RestAPI необходимо придумать и задать логин и пароль подключения в настройках TimeControl («Сервис-Настройки», как показано на рисунке ниже), которые в дальнейшем будут использоваться для подключения. После задания логина и пароля необходимо перезапустить службу «TimeControlWebService».



## Пример вызовы

Для соединения используется **HTTP Basic Authorization** с логином и паролем, заданным в настройках подключения к API в TimeControl.

Строка вызова: [**http://[IP]:5053/api/[command]?[params**](http://[IP]:5053/api/%5bcommand%5d?%5bparams)**]**

[IP] – адрес сервера TimeControl

[command] – название команды

[params] – параметры (если заданы).

Пример, <http://127.0.0.1:5053/api/testconnect> команда проверки связи.

В случае удачного подключения код состояния ответа (response code) должен быть **200** (или **204)**, тип контента (Content-Type): **application/json; charset=utf-8** и тело ответа содержать **json** результат с версией API и версией TimeControl**.**

{

    "API\_verison": "1.0",

    "TimeControl\_verison": "5.5"

}

## Возврат ошибки

Если код состояния ответа (response code) **500** (ошибка сервера) или 405 (ошибка выполнения команды), тогда тип контента (Content-Type): **application/json; charset=utf-8** и тело ответа содержат **json** результата с кодом и описание ошибки**.**

{

    "error\_code": 1

"error": "Текст ошибки"

}

error\_code – код ошибки:

|  |  |
| --- | --- |
| 1 | Ошибка парсинга json |
| 2 | Ошибка подключения к базе данных |
| 3 | Потеря связи с базой данных |
| 4 | Ошибка выполнения SQL запроса |
| 5 | Ошибка: нет данных |
| 6 | Запрещено редактировать поле |
| 7 | Нет полей для записи в базу |
| 8 | Не заполнен обязательный параметр |
| 9 | Запрещена полная очистка |
| 10 | Не найдена запись |

error – текст ошибки

## Системные структуры

Часто используемые структуры в **json** теле запроса

### Структура поиска сотрудника PersonSearch

Данная структура используется при поиске сотрудника по ключевым параметрам. В структуре должен присутствовать хотя бы один из параметров. Приоритет поиска: UID, DEVICE\_UID и затем TABNUM.

“PersonSearch” : {

    "UID": 1

"DEVICE\_UID": 1

"TABNUM": "000000-01"

}

# Сервисные функции

## Проверка связи (testconnect)

Проверка связи с API сервера TimeControl.

**Название команды:**

testconnect

**Параметры:**

отсутствуют

**Результат:**

200 OK

{

    "API\_verison": "1.0",

    "TimeControl\_verison": "5.5"

}

Где,

API\_verison – строка версия API

TimeControl\_verison – строка версия TimeControl

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/testconnect>

## Выполнение SQL запроса (select)

Получение SQL выборки из базы данных TimeControl. Возвращает выборку в виде JSON представления. Данные выборки хранятся в реквизите data. Реквизит record\_count показывает количество строк в результирующей выборке.

**Название команды:**

select

**Параметры:**

|  |  |  |
| --- | --- | --- |
| sql | Текст SQL запроса |  |

**Результат:**

**На примере вызова:** http:// 127.0.0.1:5053/api/select?sql=select \* from DEVICES

200 OK

{

    "record\_count": 5,

    "data": [

        {

            "rec\_no": 1,

            "DEVID": "92",

            "DEVNAME": "2516",

            "DEVTYPE": "102",

            "DID": "1",

            "WORKTYPE": "4",

            "IP": "",

            "ISUSE": "1",

            "COMPORT": "0",

            "COMPNAME": "DENISNEW",

            "COMBAUD": "0",

            "REPL$ID": "82",

            "REPL$GRPID": "1",

            "UPDATEMODE": "0",

            "CID": "",

            "USEWORKCODE": "",

            "CORRECTTIME": "",

            "DEFAULT\_STATUS": "",

            "CAMERA\_DELAY": "",

            "HOURDIF": "0",

            "CID2": "",

            "TRANSLIT\_MANUAL": "0",

            "BANK\_MODE": "",

            "CAMERA\_DELAY\_ADD": "",

            "LAST\_RECORD\_LOAD": "23.05.2022 16:51:40",

            "DISABLE\_IN\_MONITOR": "",

            "DEV\_SUBTYPE": "1"

        },

**Пример вызова:**

http:// 127.0.0.1:5053/api/select?sql=select \* from DEVICES

# Сотрудники

## Справочник должностей (doljnost)

Получение списка должностей, добавленных в справочник должностей.

**Название команды:**

doljnost

**Параметры:**

отсутствуют

**Результат:**

200 OK

{

    "record\_count": 2,

    "data": [

        {

            "rec\_no": 1,

            "ID": "29",

            "DOLJNAME": "Бетонщик",

            "CODE": "108"

        },

        {

            "rec\_no": 2,

            "ID": "41",

            "DOLJNAME": "Ведущий инженер",

            "CODE": ""

        }

    ]

}

Где,

record\_count – количество записей в разделе data (данные)

data – список должностей

rec\_no – число номер записи

ID – строка ID (внутренний код) должности

DOLJNAME – строка название должности

CODE – строка текстовый код должности

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/doljnost>

## Добавить должность (doljnost/add)

Добавление должности в справочник должностей.

**Название команды:**

doljnost/add

**Параметры:**

|  |  |  |
| --- | --- | --- |
| dictvalue | Наименование должности | обязательный |
| code | Текстовый код должности | не обязательный |
| pid | Идентификатор профиля учета рабочего времени | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1,

    "ID": "63"

}

Где,

record\_count – количество добавленных записей (всегда одно в данной команде)

ID – идентификатор должности в справочнике должностей

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/doljnost/add?dictvalue=Монтажник>

## Удалить должность (doljnost/delete)

Удаление должности из справочника. Если параметры не указаны, то будут удалены все записи из справочника.

**Название команды:**

doljnost/delete

**Параметры:**

|  |  |  |
| --- | --- | --- |
| dictvalue | Наименование должности | не обязательный |
| id | Идентификатор должности | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество удаленных записей

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/doljnost/delete?id=15

## Справочник подразделений (department)

Получение списка подразделений в TimeControl

**Название команды:**

department

**Параметры:**

отсутствуют

**Результат:**

200 OK

{

    "record\_count": 3,

    "data": [

        {

            "rec\_no": 1,

            "CODE": "",

            "DEPNAME": "Администрация",

            "DEPID": "1",

            "PARENTID": "0",

            "PARENT\_NAME": "",

            "PARENT\_CODE": "",

            "CHILD\_LIST": "1"

        },

        {

            "rec\_no": 2,

            "CODE": "",

            "DEPNAME": "Крон-Ц. Филиал в г. Волгоград",

            "DEPID": "173",

            "PARENTID": "175",

            "PARENT\_NAME": "ОАО \"Крон-Ц\"",

            "PARENT\_CODE": "",

            "CHILD\_LIST": "173,177,188"

        },

        {

            "rec\_no": 3,

            "CODE": "",

            "DEPNAME": "Крон-Ц. Филиал в г. Калининград",

            "DEPID": "174",

            "PARENTID": "175",

            "PARENT\_NAME": "ОАО \"Крон-Ц\"",

            "PARENT\_CODE": "",

            "CHILD\_LIST": "174,176,187"

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список подразделений

rec\_no – число номер записи

DEPID – строка ID (внутренний код) подразделения

PARENTID – строка ID (внутренний код) родительского подразделения. Если "0", то подразделение в корне.

CODE – строка текстовый код подразделения

DEPNAME – строка название подразделения

PARENT\_NAME – строка название родительского подразделения

PARENT\_CODE – строка текстовый код родительского подразделения

CHILD\_LIST – список ID через запятую вложенных подразделений (включая ID самого подразделения)

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/department>

## Добавить подразделение (department/add)

Добавление подразделения в справочник.

**Название команды:**

department/add

**Параметры:**

|  |  |  |
| --- | --- | --- |
| DEPNAME | Наименование | обязательный |
| code | Текстовый код должности | не обязательный |
| PARENTID | Идентификатор родительского подразделения | не обязательный |
| DEPART\_TYPE | 0 – подразделение 1 – организация в структуре подразделений | не обязательный |
| DEPCOMMENT | Комментарий | не обязательный |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1,

    "DEPID": "63"

}

Где,

record\_count – количество добавленных записей (всегда одно в данной команде)

DEPID – идентификатор подразделения в справочнике подразделений

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/department/add?depname=>администрация

## Удалить подразделение (department/delete)

Удаление подразделения из справочника. Если параметры не указаны, то будут удалены все записи из справочника.

**Название команды:**

department/delete

**Параметры:**

|  |  |  |
| --- | --- | --- |
| depname | Наименование | не обязательный |
| depid | Идентификатор подразделения | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество удаленных записей

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/department/delete?depid=15

## Справочник сотрудников (persons)

Получение списка сотрудников.

**Название команды:**

persons

**Параметры:**

|  |  |  |
| --- | --- | --- |
| fullname | Фильтр по ФИО | вхождение |
| uid | Фильтр по внутреннему идентификатору | полное совпадение |
| depart | Фильтр по идентификатору подразделения | полное совпадение |
| device\_uid | Фильтр по ID сотрудника на устройстве | полное совпадение |
| tabnum | Фильтр по табельному номеру | полное совпадение |
| dolj | Фильтр по идентификатору должности | полное совпадение |
| doljname | Фильтр по названию должности | вхождение |
| depname | Фильтр по названию подразделения | вхождение |
| islocked | 0 – только работающих, 1 – только уволенных |  |

**Результат:**

200 OK

{

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "UID": "1238",

            "FIRSTNAME": "Иванова",

            "SNAME": "Екатерина Михайловна",

            "MIDNAME": "",

            "FULLNAME": "Иванова Екатерина Михайловна",

            "DEVICE\_UID": "8",

            "TABNUM": "ТДЗП-20534",

            "DEPART": "3",

            "POL": "1",

            "MOBPHONE": "8910111111",

            "ISLOCKED": "1",

            "LOCKDATE": "24.03.2020",

            "DOLJ": "17",

            "SHEDCARDKEY": "",

            "BITHDAY": "23.06.1967",

            "STARTDATE": "",

            "DOLJNAME": "Электромонтажник",

            "DEPNAME": "Склад магазин"

        }

]

}

Где

record\_count – количество записей в разделе data (данные)

data – список сотрудников

rec\_no – число номер записи

UID – строка ID (внутренний код) сотрудника

FULLNAME – строка ФИО

DEVICE\_UID – строка ID (внутренний код) сотрудника на устройстве

DEPART – строка ID (внутренний код) подразделения сотрудника

DOLJ – строка ID (внутренний код) должности сотрудника

TABNUM - строка табельный номер

POL – 1 – мужской 2 – женский

MOBPHONE – мобильный телефон

ISLOCKED – 0 - работает 1 – уволен

LOCKDATE – дата увольнения

SHEDCARDKEY – код карты

BITHDAY – дата рождения

STARTDATE – дата приема на работу

DOLJNAME – наименование должности

DEPNAME – наименование подразделения

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons?depart=7&doljname=монтаж>

## Добавить сотрудника (persons/add)

Добавление сотрудника в справочник.

**Название команды:**

persons/add

**Параметры:**

|  |  |  |
| --- | --- | --- |
| FULLNAME | Наименование | обязательный |
| TABNUM | Табельный номер | не обязательный |
| POL | 1 – мужской 2 – женский | не обязательный |
| DEVICE\_UID | Код сотрудника на устройстве | не обязательный (автоматически сгенерируется при отсуствии) |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1,

    "UID": "63"

    "DEVICE\_UID": "5"

}

Где,

record\_count – количество добавленных записей (всегда одно в данной команде)

UID – идентификатор сотрудника

DEVICE\_UID – идентификатор сотрудника на устройстве

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/add?fullname=Иванов>

## Редактировать сотрудника (persons/edit)

Редактирование сотрудника в справочнике.

**Название команды:**

persons/edit

**Параметры:**

|  |  |  |
| --- | --- | --- |
| UID | Идентификатор сотрудника в базе данных | или |
| DEVICE\_UID | ID сотрудника на устройстве | или |
| TABNUM | Табельный номер сотрудника | или |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество редактируемых записей. Если 0 – то не найден сотрудник по идентификаторы, 1 – если сотрудник был обновлен

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/edit?uid=5&fullname=Иванов>

<http://127.0.0.1:5053/api/persons/edit?device_uid=1>

<http://127.0.0.1:5053/api/persons/edit?tabnum=’00001>’

## Увольнение\восстановление сотрудника (persons/edit)

Редактирование сотрудника в справочнике.

**Название команды:**

persons/edit

**Параметры:**

|  |  |  |
| --- | --- | --- |
| UID | Идентификатор сотрудника в базе данных | или |
| DEVICE\_UID | ID сотрудника на устройстве | или |
| TABNUM | Табельный номер сотрудника (задается в ‘’) | или |
| ISLOCKED | 1 – волен, 0 - работает | обязательный |
| LOCKEDATE | Дата увольнения | обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество редактируемых записей. Если 0 – то не найден сотрудник по идентификаторы, 1 – если сотрудник был обновлен

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/edit?uid=5&islocked=>1&lockdate=21.06.2022

## Удалить сотрудника (persons/delete)

Удаление сотрудника из справочника.

**Название команды:**

persons/delete

**Параметры:**

|  |  |  |
| --- | --- | --- |
| uid | Идентификатор сотрудника | не обязательный |
| Device\_uid | Идентификатор сотрудника на устройстве | не обязательный |
| TABNUM | Табельный номер сотрудника | не обязательный |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество удаленных записей

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/delete?uid=15>

## Получить фото сотрудника (persons/foto/get)

Получить фото сотрудника из справочника.

**Название команды:**

persons/foto/get

**Параметры:**

|  |  |  |
| --- | --- | --- |
| UID | Идентификатор сотрудника в базе данных | или |
| DEVICE\_UID | ID сотрудника на устройстве | или |
| TABNUM | Табельный номер сотрудника | или |
| FULLNAME | Поиск по вхождению ФИО сотрудника | или |
| MOBPHONE | Поиск по вхождению моб. телефона сотрудника | или |
| nun | Номер фотографии | не обязательный |
| small | 1 – возвращает маленькое фото | не обязательный |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

Тип контента: image/jpeg

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/foto/get?uid=15>

<http://127.0.0.1:5053/api/persons/foto/get?device_uid=2>

## Загрузить фото сотрудника (persons/foto/set)

Загрузка фото сотрудника. Загрузка может быть как из файла, так и из данных в теме запроса. Если файл не указан и тело запроса отсутствует, но задан параметр **facedeep**, то происходит только обновление признака использования фотографии для терминалов распознавания геометрии лица.

**Название команды:**

persons/foto/set

**Параметры:**

|  |  |  |
| --- | --- | --- |
| UID | Идентификатор сотрудника в базе данных | или |
| DEVICE\_UID | ID сотрудника на устройстве | или |
| TABNUM | Табельный номер сотрудника | или |
| FULLNAME | Поиск по вхождению ФИО сотрудника | или |
| MOBPHONE | Поиск по вхождению моб. телефона сотрудника | или |
| nun | Номер фотографии (если не указан, то 1) | не обязательный |
| facedeep | 1 – использовать фото для распознавания геометрии лица | не обязательный |
| file | Путь к файлу фотографии (если не указан, то фото берется из параметра PHOTO\_DATA) | не обязательный |

**Тело запроса (может отсутствовать)**

{

    " PHOTO\_DATA": "==DFGSDFGSDFGSDFG4T34G34GRG………."

}

**Результат:**

200 OK

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/foto/set?uid=15&file=c:\foto.jpg>

<http://127.0.0.1:5053/api/persons/foto/set?device_uid=2&facedeep=1>

## Удалить фото сотрудника (persons/foto/delete)

Удаление фото сотрудника из справочника.

**Название команды:**

persons/foto/delete

**Параметры:**

|  |  |  |
| --- | --- | --- |
| UID | Идентификатор сотрудника в базе данных | или |
| DEVICE\_UID | ID сотрудника на устройстве | или |
| TABNUM | Табельный номер сотрудника | или |
| FULLNAME | Поиск по вхождению ФИО сотрудника | или |
| MOBPHONE | Поиск по вхождению моб. телефона сотрудника | или |
| nun | Номер фотографии | не обязательный |
| facedeep | 1 – использовать фото для распознавания геометрии лица | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество удаленных записей

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/foto/delete?uid=15>

# Выручка

## Выручка из журнала «Заработная плата - Выручка» (revenue)

Получение данных из журнала выручки.

**Название команды:**

revenue

**Параметры:**

Все параметры не обязательны. Если ни один параметр не указан, то будет возвращен весь журнал.

|  |  |  |
| --- | --- | --- |
| fullname | Фильтр по ФИО | вхождение |
| uid | Фильтр по внутреннему идентификатору | полное совпадение |
| depart | Фильтр по идентификатору подразделения | полное совпадение |
| device\_uid | Фильтр по ID сотрудника на устройстве | полное совпадение |
| tabnum | Фильтр по табельному номеру | полное совпадение |
| mobphone | Фильтр по мобильному номеру сотрудника | вхождение |
| dep\_code | Фильтр по текстовому коду подразделения | полное совпадение |
| depname | Фильтр по названию подразделения | вхождение |
| is\_import | 0 – только добавленные из TimeControl, 1 – только добавленные из RestAPI | не обязательный |
| rtype | 1 – выручка сотрудника, 2 – выручка подразделения | не обязательный |
| rperiodtype | 1 – дневная выручка, 2 - месячная | не обязательный |
| startdate | Дата выручки больше или равна указанной границе | не обязательный |
| enddate | Дата выручки меньше или равна указанной границе | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 2,

    "data": [

        {

            "rec\_no": 1,

"RID": "11",

            "UID": "1188",

            "DEPID": "",

            "RCOMMENT": "",

            "ADDUID": "1188",

            "RTYPE": "1",

            "RPERIODTYPE": "2",

            "RVALUE": "5000",

            "RDATE": "01.07.2021",

            "IS\_IMPORT": "0",

            "DEPNAME": "",

            "DEP\_CODE": "",

            "FULLNAME": "Петрова Екатерина Ивановна",

            "DEVICE\_UID": "1",

            "TABNUM": "1",

            "MOBPHONE": ""

        },

        {

            "rec\_no": 2,

"RID": "12",

            "UID": "1188",

            "DEPID": "",

            "RCOMMENT": "",

            "ADDUID": "1188",

            "RTYPE": "1",

            "RPERIODTYPE": "2",

            "RVALUE": "2000",

            "RDATE": "01.07.2021",

            "IS\_IMPORT": "0",

            "DEPNAME": "",

            "DEP\_CODE": "",

            "FULLNAME": "Петрова Екатерина Ивановна",

            "DEVICE\_UID": "1",

            "TABNUM": "1",

            "MOBPHONE": ""

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список сотрудников

rec\_no – число номер записи

RID – строка ID (внутренний код) выручки

UID – строка ID (внутренний код) сотрудника

DEPID – строка ID (внутренний код) подразделения

RCOMMENT – комментарий к выручке

ADDUID – строка ID (внутренний код) сотрудника, добавившего выручку в справочник

RTYPE – 1 – выручка сотрудника, 2 - выручка подразделения

RPERIODTYPE – 1 – дневная, 2 - месячная

RVALUE – значение выручки

RDATE – дата выручки

IS\_IMPORT – 0 – добавлена из TimeControl, 1 – добавлена из RestAPI

FULLNAME – строка ФИО

DEVICE\_UID – строка ID (внутренний код) сотрудника на устройстве

DEP\_CODE – текстовый код подразделения

TABNUM - строка табельный номер

MOBPHONE – мобильный телефон

DEPNAME – наименование подразделения

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/revenue?FULLNAME=Петрова&start\_date=01.07.2021&end\_date=01.07.2021

## Добавить выручку в журнал (revenue/add)

Добавление выручки в журнал.

**Название команды:**

revenue/add

**Параметры:**

|  |  |  |
| --- | --- | --- |
| RTYPE | 1 – по сотруднику, 2 – по подразделению | обязательный |
| UID | Идентификатор сотрудника в базе данных | или (для rtype=1) |
| DEVICE\_UID | ID сотрудника на устройстве | или (для rtype=1) |
| TABNUM | Табельный номер сотрудника | или (для rtype=1) |
| FULLNAME | Поиск по вхождению ФИО сотрудника | или (для rtype=1) |
| MOBPHONE | Поиск по вхождению моб. телефона сотрудника | или (для rtype=1) |
| DEPID | Идентификатор подразделения | или (для rtype=2) |
| DEP\_CODE | Текстовый код подразделения | или (для rtype=2) |
| DEPNAME | Название подразделения (поиск по вхождению) | или (для rtype=2) |
| RVALUE | Размер выручки | обязательный |
| RPERIODTYPE | 1 – дневная (по умолчанию), 2 - месячная | не обязательный |
| RDATE | Дата выручки (по умолчанию текущая дата) | не обязательный |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1,

    "RID": "100",

    "UID": "1238",

"DEPID": "0",

    "RTYPE": "1"

}

Где,

record\_count – количество добавленных записей (всегда одно в данной команде)

RID – идентификатор выручки

UID – идентификатор сотрудника

DEPID – идентификатор подразделения

RTYPE – 1 – по сотруднику, 2 – по подразделению

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/revenue/add?rtype=1&tabnum=ДЗП-20534&rvalue=1000.50>

## Удалить выручку из журнала (revenue/delete)

Добавление выручки в журнал.

**Название команды:**

revenue/delete

**Параметры фильтры:**

|  |  |  |
| --- | --- | --- |
| RTYPE | 1 – по сотруднику, 2 – по подразделению | не обязательный |
| UID | Идентификатор сотрудника в базе данных | или (для rtype=1) **не** обязательный |
| DEVICE\_UID | ID сотрудника на устройстве | или (для rtype=1) **не** обязательный |
| TABNUM | Табельный номер сотрудника | или (для rtype=1) **не** обязательный |
| FULLNAME | Поиск по вхождению ФИО сотрудника | или (для rtype=1) **не** обязательный |
| MOBPHONE | Поиск по вхождению моб. телефона сотрудника | или (для rtype=1) **не** обязательный |
| DEPID | Идентификатор подразделения | или (для rtype=2) **не** обязательный |
| DEP\_CODE | Текстовый код подразделения | или (для rtype=2) **не** обязательный |
| DEPNAME | Название подразделения (поиск по вхождению) | или (для rtype=2) **не** обязательный |
| RVALUE | Размер выручки | **не** обязательный |
| RPERIODTYPE | 1 – дневная (по умолчанию), 2 - месячная | не обязательный |
| RDATE | Дата выручки | не обязательный |
| IS\_IMPORT | 1 – если нужно удалить только добавленные через RestAPI | не обязательный |
| (и другие поля из справочника) |  | не обязательный |

**Результат:**

200 OK

{

    "record\_count": 1

}

Где,

record\_count – количество удаленных записей

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/revenue/delete?tabnum=ДЗП-20534&rdate=21.02.2024

# События регистрации

## События регистрации за период (fact\_events)

Получение списка событий регистраций сотрудников.

**Название команды:**

fact\_events

**Параметры:**

|  |  |  |
| --- | --- | --- |
| fullname | Фильтр по ФИО | вхождение |
| uid | Фильтр по идентификатору сотрудника | полное совпадение |
| doorid | Фильтр по идентификатору проходной | полное совпадение |
| device\_uid | Фильтр по ID сотрудника на устройстве | полное совпадение |
| useforfact | 1 – события учета рабочего времени, 0 – событие прохода | полное совпадение |
| inouttype | Тип события: 0 - не задан (авто) 1 - на вход, 2 - на выход | полное совпадение |
| regmetod | Метод регистрации события: 1 - вручную 2 - картой 3 - автоматически 4 - биометрия | полное совпадение |
| startdate | События старше указанной даты | >= |
| enddate | События младше указанной даты | <= |

**Результат:**

200 OK

{

    "record\_count": 2,

    "data": [

        {

            "rec\_no": 1,

            "REGID": "5744",

            "UID": "3907",

            "DEVICE\_UID": "519",

            "FULLNAME": "Бажова Светлана Нурисламовна",

"TABNUM": "0001",

            "REGDATE": "11.01.2023 19:00:00",

            "REGDATEFULL": "11.01.2023 19:00:00",

            "DOORID": "1",

"DOOR\_CODE": "01",

            "REGMETOD": "1",

            "USEFORFACT": "1",

            "INOUTTYPE": "0",

            "WORKCOUNT": "0"

        },

        {

            "rec\_no": 2,

            "REGID": "5745",

            "UID": "3907",

            "DEVICE\_UID": "519",

            "FULLNAME": "Бажова Светлана Нурисламовна",

"TABNUM": "0001",

            "REGDATE": "11.01.2023 23:00:00",

            "REGDATEFULL": "11.01.2023 23:00:00",

            "DOORID": "1",

"DOOR\_CODE": "01",

            "REGMETOD": "1",

            "USEFORFACT": "1",

            "INOUTTYPE": "0",

            "WORKCOUNT": "0"

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список событий

rec\_no – число номер записи

UID – строка ID (внутренний код) сотрудника

REGID – строка ID (внутренний код) события

FULLNAME – строка ФИО

DEVICE\_UID – строка ID (внутренний код) сотрудника на устройстве

REGDATE – дата и время события округленное до минут

REGDATEFULL – дата и время события с секундами

DOORID – строка ID (внутренний код) проходной

REGMETOD - метод регистрации события: 1 - вручную 2 - картой 3 - автоматически 4 - биометрия

USEFORFACT - 1 – события учета рабочего времени, 0 – событие прохода

INOUTTYPE - тип события: 0 - не задан (авто) 1 - на вход, 2 - на выход

WORKCOUNT – количество единиц (при использовании функции учета операций)

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

События за 11.01.2023:

<http://127.0.0.1:5053/api/fact_events?startdate=11.01.2023&enddate=11.01.2023>

## Добавить событие регистрации (fact\_events/add)

Добавление сотрудника в справочник.

**Название команды:**

fact\_events/add

**Параметры:**

|  |  |  |  |
| --- | --- | --- | --- |
| uid | Идентификатор сотрудника в БД | Обязательный или device\_uid |  |
| device\_uid | ID сотрудника на устройстве | Обязательный или uid |  |
| add\_uid | Идентификатор сотрудника зарегистрировавшего событие | не обязательный |  |
| event\_datetime | Дата и время события в формате 01.01.2023 12:00 | не обязательный | Если не указан, то текущая дата и время |
| doorid | Идентификатор проходной | не обязательный | Если не указан, то проходная по умолчанию при ручном добавлении |
| inouttype | 0 – авто, 1 – событие входа, 2 – событие ухода | не обязательный | 0 |
| useforfact | 1 – событие влияет на учет рабочего времени, 0 – событие факта прохода через проходную без учета | не обязательный | 1 |

**Результат:**

200 OK

{

    "result": 1,

    "ADD\_RESULT": "1"

}

Где,

result – результат выполнения операции в БД. 1 – УСПЕШНО.

ADD\_RESULT – результат добавления события регистрации:

0 - ошибка

1 - зарегистрирован приход

2 - зарегистрирован уход

3 - отметка уже есть

4 - зарегистрирован проход без учета рабочего времени

5 - ошибка есть более поздние интервалы

6 - пересчет

7 - неверная карта

8 - только добавление записи

9 - ошибка установки

10 - событие старее 1 года

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/fact_events/add?uid=>1188&event\_datetime=07.02.2023 11:00

# Учет рабочего времени

## Интервалы присутствия (time/intervals)

Получение списка интервалов присутствия сотрудников.

**Название команды:**

time/intervals

**Параметры:**

|  |  |  |
| --- | --- | --- |
| fullname | Фильтр по ФИО | вхождение |
| uid | Фильтр по идентификатору сотрудника | полное совпадение |
| adddoorid | Фильтр по идентификатору проходной входа | полное совпадение |
| closedoorid | Фильтр по идентификатору проходной выхода | полное совпадение |
| device\_uid | Фильтр по ID сотрудника на устройстве | полное совпадение |
| addtype | Тип открытия интервала: 1 - вручную, 2 - картой, 3 - автоматически, 4 - биометрия, 5 - пароль | полное совпадение |
| closetype | Тип закрытия интервала: 1 - вручную, 2 - картой, 3 - автоматически, 4 - биометрия, 5 - пароль | полное совпадение |
| isgo | 1 – текущие интервалы (нет ухода), 0 – законченные интервалы (есть приход и уход) | полное совпадение |
| startdate | Интервалы старше указанной даты | >= |
| enddate | Интервалы младше указанной даты | <= |

**Результат:**

200 OK

{

    "record\_count": 2,

    "data": [

        {

            "rec\_no": 1,

            "DEVICE\_UID": "12",

            "UID": "1383",

            "STARTTIME": "05.07.2022 16:15:00",

            "ENDTIME": "05.07.2022 16:16:00",

            "ISGO": "0",

            "ADDTYPE": "4",

            "CLOSETYPE": "4",

            "ADDDOOR": "1",

            "CLOSEDOOR": "1",

            "STARTDATE": "05.07.2022",

            "ENDDATE": "05.07.2022"

        },

        {

            "rec\_no": 2,

            "DEVICE\_UID": "3",

            "UID": "1190",

            "STARTTIME": "05.07.2022 11:43:00",

            "ENDTIME": "05.07.2022 18:00:00",

            "ISGO": "0",

            "ADDTYPE": "2",

            "CLOSETYPE": "3",

            "ADDDOOR": "1",

            "CLOSEDOOR": "1",

            "STARTDATE": "05.07.2022",

            "ENDDATE": "05.07.2022"

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список событий

rec\_no – число номер записи

UID – строка ID (внутренний код) сотрудника

FULLNAME – строка ФИО

DEVICE\_UID – строка ID (внутренний код) сотрудника на устройстве

STARTTIME – дата и время начала интервала

ENDTIME – дата и время окончания интервала

ADDTYPE – тип открытия интервала: 1 - вручную, 2 - картой, 3 - автоматически, 4 - биометрия, 5 - пароль CLOSETYPE - тип закрытия интервала: 1 - вручную, 2 - картой, 3 - автоматически, 4 - биометрия, 5 - пароль

ISGO - 1 – текущий интервал (нет ухода), 0 – законченный интервал

ADDDOOR - идентификатору проходной входа

CLOSEDOOR – идентификатору проходной выхода

STARTDATE – дата начала интервала

ENDDATE – дата окончания интервала

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

Интервалы за 11.01.2023:

<http://127.0.0.1:5053/api/time/intervals?startdate=11.01.2023&enddate=11.01.2023>

## Присутствующие на данный момент сотрудники (time/intervals)

Получения списка присутствующих на данный момент сотрудников с помощью команды time/intervals

**Параметры:**

|  |  |  |
| --- | --- | --- |
| isgo | 1 – текущие интервалы (нет ухода), 0 – законченные интервалы (есть приход и уход) | полное совпадение |

**Пример вызова:**

http://127.0.0.1:5053/api/time/intervals?isgo=1

## Получение рабочего времени (план\факт) по дням (time/works)

Получение рабочего времени сотрудников планового и фактического с учетом опозданий, ранних уходов, выполнением графика.

**Название команды:**

time/works

**Параметры:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| startdate | Интервалы старше указанной даты | >= | обязательный |  |
| enddate | Интервалы младше указанной даты | <= | обязательный |  |
| depart\_list | Фильтр по подразделениям. Список идентификаторов подразделений через , | полное совпадение | не обязательный |  |
| person\_list | Фильтр по подразделениям. Список идентификаторов (UID) сотрудников через , | полное совпадение | не обязательный |  |
| device\_uid | Фильтр по ID сотрудника на устройстве | полное совпадение | не обязательный |  |
| Use\_isgo | Если 0, то только закрытый день (приход и уход), 1 – все дни | полное совпадение |  | 1 |

**Результат:**

200 OK

{

    "record\_count": 2,

    "data": [

        {

            "rec\_no": 1,

            "FULLNAME": "Петрова Екатерина Ивановна",

            "TABNUM": "1",

            "DEVICE\_UID": "1",

            "DEPART": "3",

            "UID": "1188",

            "WORKDATE": "05.07.2022",

            "START\_TIME": "05.07.2022 9:00:00",

            "END\_TIME": "05.07.2022 18:00:00",

            "MIN\_WORK": "540",

            "DAYTYPE": "0",

            "ADDDAYTYPE": "0",

            "FACT\_START\_TIME": "",

            "FACT\_END\_TIME": "",

            "FACT\_MIN\_WORK": "",

            "OPOZD": "0",

            "ZADERJ": "0",

            "EARLY\_OUT": "0",

            "WORK\_DIFIRENS": "540",

            "PROGUL": "1",

            "ISGO": "0",

            "DAYNAME\_SHORT": "Я"

        },

        {

            "rec\_no": 2,

            "FULLNAME": "Кошелева Анастасия Алексеевна",

            "TABNUM": "2",

            "DEVICE\_UID": "3",

            "DEPART": "3",

            "UID": "1190",

            "WORKDATE": "05.07.2022",

            "START\_TIME": "05.07.2022 9:00:00",

            "END\_TIME": "05.07.2022 18:00:00",

            "MIN\_WORK": "540",

            "DAYTYPE": "0",

            "ADDDAYTYPE": "0",

            "FACT\_START\_TIME": "05.07.2022 11:43:00",

            "FACT\_END\_TIME": "05.07.2022 18:00:00",

            "FACT\_MIN\_WORK": "377",

            "OPOZD": "163",

            "ZADERJ": "0",

            "EARLY\_OUT": "0",

            "WORK\_DIFIRENS": "163",

            "PROGUL": "0",

            "ISGO": "0",

            "DAYNAME\_SHORT": "Я"

        },

Где

record\_count – количество записей в разделе data (данные)

data – список дней

rec\_no – число номер записи

UID – строка ID (внутренний код) сотрудника

FULLNAME – строка ФИО

TABNUM - строка табельный номер сотрудника

DEVICE\_UID – строка ID (внутренний код) сотрудника на устройстве

DEPART – строка ID (внутренний код) подразделения сотрудника

WORKDATE – дата смены

START\_TIME – дата и время начала смены по графику

END\_TIME – дата и время окончания смены по графику

MIN\_WORK – продолжительность смены в минутах по графику

DAYTYPE – тип дня: 0 – рабочий день, 1 – больничный, 2 – отпуск, 3 – командировка, 4 – отгул, 5 – прогул

FACT\_START\_TIME – дата и время начала смены по факту

FACT\_END\_TIME - дата и время окончания смены по факту

FACT\_MIN\_WORK – продолжительность смены в минутах по факту

OPOZD – опоздание в минутах

ZADERJ – задержка в минутах

EARLY\_OUT – ранний уход в минутах

WORK\_DIFIRENS – отклонение от графика в минутах

PROGUL – 1 – прогул

ISGO – 1 – день не закрыт (нет ухода, на работе на данный момент)

DAYNAME\_SHORT – сокращенное наименования дня

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

Интервалы за 11.01.2023:

<http://127.0.0.1:5053/api/time/works?startdate=11.01.2023&enddate=11.01.2023>

# Импорт данных в TimeControl (в том числе для обмена с 1С)

## Импорт должностей списком (doljnost/addfromlist)

Загрузка списка должностей в TimeControl. Уникальность записи проверяется по названию должности. Если должность с таким названием есть в TimeControl, то она не будет добавлена.

**Название команды:**

doljnost/addfromlist

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    "record\_count": 2,

    "data": [

        {

            "rec\_no": 1,

            "Наименование": "Бетонщик",

            "Источн**ик":** 2

        },

        {

            "rec\_no": 2,

            "Наименование": "Секретарь",

            "Источн**ик":** 2

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список должностей для загрузки

rec\_no – число номер записи

Наименование – название должности

Источн**ик –** код источника (1 – при импорте из 1С, иначе можно любой свой код: 2,3 ... и т.д.)

**Результат:**

200 OK

{

    "allcount": 2,

    "addcount": 2,

    "beforeadd": 0

}

Где

allcount – количество записей, поступивших на загрузку

addcount – добавлено новых записей

beforeadd – ранее добавлено (запись из списка на загрузку, которая уже существовала в TimeControl)

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/doljnost/addfromlist

## Импорт справочника юридических лиц списком (company/addfromlist)

Загрузка списка юридических лиц в TimeControl. Уникальность записи проверяется по ИНН организации. Если организация с таким ИНН есть в TimeControl, то она не будет добавлена, а только обновлена в случае изменения каких-либо реквизитов.

**Название команды:**

company/addfromlist

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "Наименование": "ООО Рога и Копыта",

            "НаименованиеПолное": " Общество с ограниченной ответственностью Рога и Копыта",

            "ИНН": "123456789",

            "КПП": "111111111",

            "ОГРН": "",

            "ГоловнаяИНН": "",

            "Руководитель": "",

            "ДолжностьРуководителя": "",

            "ГлавныйБухгалтер": "",

            "Источн**ик":** 2

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список организаций для загрузки

rec\_no – число номер записи

Наименование – наименование организации

НаименованиеПолное – полное наименование организации

ИНН – ИНН организации

КПП – КПП организации

ОГРН – ОГРН организации

ГоловнаяИНН – ИНН головной организации

Руководитель – ФИО руководителя организации

ДолжностьРуководителя – должность руководителя организации

ГлавныйБухгалтер – ФИО главного бухгалтера организации

Источн**ик –** код источника (1 – при импорте из 1С, иначе можно любой свой код: 2,3 ... и т.д.)

**Результат:**

200 OK

{

    "allcount": 1,

    "addcount": 1,

    "beforeadd": 0,

    "updatecount": 0

}

Где

allcount – количество записей, поступивших на загрузку

addcount – добавлено новых записей

beforeadd – ранее добавлено (запись из списка на загрузку, которая уже существовала в TimeControl)

updatecount – количество обновленных записей (запись с ключевым полем присутствовала, но были изменены и обновлены реквизиты)

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/company/addfromlist

## Импорт подразделений списком (department/addfromlist)

Загрузка списка подразделений в TimeControl. Уникальность записи проверяется по ИНН организации (если Тип=1 организация в структуре подразделений) или по текстовому коду подразделения + ИНН организации (если Тип=0 подразделение в рамках организации). Если запись уникальна, то она будет добавлена, а если нет, то обновлена при изменении реквизитов.

**Название команды:**

department/addfromlist

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "Наименование": "Администрация",

            "ОрганизацияИНН": "12345678",

            "Код": "00001",

            "КодРодителя": "0",

            "Тип": 0,

            " ОрганизацияПолное": "",

            " ОрганизацияСокрощенно": "",

            "Руководитель": "",

            "ГлавныйБухгалтер": "",

            "Источн**ик":** 2

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список подразделений для загрузки

rec\_no – число номер записи

Наименование – наименование подразделения

ОрганизацияИНН – ИНН организации, к которой принадлежит подразделение. Не может быть пустым. Если структура организации не используется, укажите произвольный. К примеру, «1»

Код – текстовый код организации. Не может быть пустым и должен быть уникальным в рамках одной организации

КодРодителя – текстовый код родительского подразделения для построения иерархии

Тип – 0 - если подразделение, 1 - если организация в структуре подразделений

ОрганизацияПолное – название организации, к которой принадлежит подразделение

ОрганизацияСокрощенно – название организации (сокращенное), к которой принадлежит подразделение

Руководитель – ФИО руководителя организации

ГлавныйБухгалтер – ФИО главного бухгалтера организации

Источн**ик –** код источника (1 – при импорте из 1С, иначе можно любой свой код: 2,3 ... и т.д.)

**Результат:**

200 OK

{

    "allcount": 1,

    "addcount": 1,

    "beforeadd": 0,

    "updatecount": 0

}

Где

allcount – количество записей, поступивших на загрузку

addcount – добавлено новых записей

beforeadd – ранее добавлено (запись из списка на загрузку, которая уже существовала в TimeControl)

updatecount – количество обновленных записей (запись с ключевым полем присутствовала, но были изменены и обновлены реквизиты)

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/depart/addfromlist

## Импорт сотрудников списком (persons/addfromlist)

Загрузка сотрудников в TimeControl. Уникальность записи проверяется по ФИО + дата рождения или по коду физ или по ФИО или по Табельному номеру. лица в зависимости от настройки «МетодПоиска». Если запись уникальна, то она будет добавлена, а если нет, то обновлена при изменении реквизитов.

**Название команды:**

persons/addfromlist

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    "МетодПоиска": "ФИОиДата",

    "ДобавлятьУволенных": false,

    "ЗагружатьТелефоны": true,

    "ЗагружатьАдрес": false,

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "ФИО": "Иванов Иван Иванович",

            "Фамилия": "Иванов",

            "Имя": "Иван",

            "Отчество": "Иванович",

            "Пол": "Мужской",

            "ДатаРождения": "21.06.1984",

            "ФизическоеЛицоКод": "0059",

            "АдресМестаПроживания": "",

            "Подразделение": "Администрация",

            "ПодразделениеКод": "00001",

            "Организация": "ООО Рога и Копыта",

            "ОрганизацияИНН": "12345678",

            "ГоловнаяИНН": "",

            "Должность": "Директор",

            "ТабельныйНомер": "000001",

            "ДатаПриема": "01.01.2020",

            "ДатаУвольнения": "",

            "ТелефонРабочий": "",

            "ТелефонМобильный": "",

            "ВидЗанятости": "",

            "ВидЗанятостиКод": "",

            "ВидСобытия": "",

            "ВидСобытияКод": "",

            "Источн**ик":** 2

        }

    ]

}

Где,

МетодПоиска – если “ФИОиДата”, то уникальность записи проверяется по ФИО + ДатаРождения, если “КодФизЛица”, то уникальность проверяется по ФизическоеЛицоКод, если “ФИО”, то уникальность записи проверяется только по ФИО, если “ТабельныйНомер”, то уникальность записи проверяется только по ТабельныйНомер.

ДобавлятьУволенных – если true, то сотрудники с ВидСобытия «Увольнение» также будут загружаться в TimeControl со статусом уволенный и датой увольнения. Иначе они не загружаются

ЗагружатьТелефоны – если true, то будет загружаться в TimeControl ТелефонРабочий и ТелефонМобильный

ЗагружатьАдрес – если true, то будет загружаться в TimeControl АдресМестаПроживания

record\_count – количество записей в разделе data (данные)

data – список сотрудников для загрузки

rec\_no – число номер записи

ФИО – ФИО сотрудника

Пол – могут быть указаны значения «Мужской» или «Женский»

ДатаРождения – дата рождения

ФизическоеЛицоКод – строка уникальный код сотрудника. Может не использоваться при методе поиска ФИОиДата

АдресМестаПроживания – текстовый адрес

Подразделение – название подразделение (можно не указывать)

ПодразделениеКод – текстовый код подразделения. Необходимо заполнять для корректной привязки сотрудника к подразделению

Организация – название юридического лица (можно не указывать)

ОрганизацияИНН – ИНН организации (юридического лица), к которой принадлежит сотрудник. Если структура организации не используется, укажите произвольный. К примеру, «1»

ГоловнаяИНН – ИНН головной организации (можно не указывать)

Должность – наименование должности сотрудника

ТабельныйНомер – табельный номер сотрудника

ДатаПриема – дата приема сотрудника на работу

ДатаУвольнения – дата увольнения сотрудника. Если заполнено и ВидСобытия указан «Увольнение», то сотрудник в TimeControl будет уволен (помечен как уволенный)

ВидЗанятости и ВидЗанятостиКод – не используются на данный момент

ВидСобытия – если указано «Увольнение», то сотрудник в TimeControl будет уволен (помечен как уволенный)

ВидСобытияКод – не используются на данный момент

Источн**ик –** код источника (1 – при импорте из 1С, иначе можно любой свой код: 2,3 ... и т.д.)

**Результат:**

200 OK

{

    "allcount": 1,

    "addcount": 1,

    "beforeadd": 0,

    "updatecount": 0,

    "lockcount": 0,

    "unlockcount": 0,

    "changedepart": 0,

    "changedolj": 0,

    "changeorg": 0,

    "message": “”,

    "break": false,

}

Где

allcount – количество записей, поступивших на загрузку

addcount – добавлено новых записей

beforeadd – ранее добавлено (запись из списка на загрузку, которая уже существовала в TimeControl)

updatecount – количество обновленных записей (запись с ключевым полем присутствовала, но были изменены и обновлены реквизиты)

lockcount – количество сотрудников, у которых поменялся статус с работающего на уволенного

unlockcount – количество сотрудников, восстановленных из уволенных

changedepart – количество сотрудников, у которых сменилось подразделение

changedolj – количество сотрудников, у которых сменилась должность

changeorg – количество сотрудников, у которых сменилось юридическое лицо

message – текст сообщения, если операция была прервана

break – если true, то операция была прервана и не все сотрудники из списка обработаны. В message будет указана причина

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

<http://127.0.0.1:5053/api/persons/addfromlist>

## Импорт отсутствий списком (absences/addfromlist)

Загрузка отсутствий (больничных, отпусков, командировок и т.д.) в TimeControl. Уникальность записи проверяется по Дате документа и типу (больничный, командировка, отпуск, отгул и прогул). К примеру, если уже есть больничный у сотрудника с той же датой документа, то отсутствие не будет добавлено, а обновлено при изменении реквизитов.

Если за период загрузки отсутствие есть в TimeControl, но его нет в списке на загрузку, то оно будет удалено из TimeControl.

**Название команды:**

absences/addfromlist

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    " ДатаНачала": "01.01.2023",

    " ДатаОкончания": "31.01.2023",

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "Наименование": "больничный сотрудника",

            "ДатаДокумента": "10.01.2023",

            "ДатаНачала": "10.01.2023",

            "ДатаОкончания": "13.01.2023",

            "Внутрисменны": false,

            "ЧасовВнутриСмены": 0,

            "БуквенныйКод": "Б",

            "ЦифровойКод": "19",

            "Номер": "0001",

            "Тип": 1,

            "Тип1С": 2,

            "Комментарий": "",

            "ФИО": "Иванов Иван Иванович",

            "ТабельныйНомер": "000001",

            "ФизическоеЛицоКод": "0059",

            "ОрганизацияИНН": "12345678",

            "Источн**ик":** 2

        }

    ]

}

Где

ДатаНачала и ДатаОкончания - начало и конец периода загрузки документов отсутствий

record\_count – количество записей в разделе data (данные)

data – список отсутствий для загрузки дата документа которых в периоде загрузки

rec\_no – число номер записи

ФИО – ФИО сотрудника (можно не указывать)

ДатаДокумента – дата документа

ДатаНачала – начало периода отсутствия

ДатаОкончания – конец периода отсутствия

ФизическоеЛицоКод – (можно не указывать, если при загрузке сотрудников не использовался)

Внутрисменны – если true, то отсутствие считается внутристенным

ЧасовВнутриСмены – указывается количество часов для внутрисменного отсутствия

БуквенныйКод – буквенный код отсутствия. По нему происходит сопоставления загружаемого отсутствия со справочником типов дней графика работ в TimeControl

ЦифровойКод – цифровой код отсутствия (можно не указывать)

ОрганизацияИНН – ИНН организации (юридического лица), к которой принадлежит сотрудник. Если структура организации не используется, укажите произвольный. К примеру, «1»

Тип – 1 – больничный, 2 – отпуск, 3 – командировка, 4 – отгул, 5 – прогул (обязательно для заполнения)

Тип1С – тип отсутствия в 1С (можно не указывать)

Номер – номер документа отсутствия (можно не указывать)

ТабельныйНомер – табельный номер сотрудника (обязательный для заполнения). По ТабельныйНомер, ФизическоеЛицоКод и ОрганизацияИНН осуществляется поиск сотрудника в TimeControl

Источн**ик –** код источника (1 – при импорте из 1С, иначе можно любой свой код: 2,3 ... и т.д.)

**Результат:**

200 OK

{

    "allcount": 1,

    "addcount": 1,

    "beforeadd": 0,

    "updatecount": 0,

    "deletecount": 0,

    "personnotfound": 0

}

Где

allcount – количество записей, поступивших на загрузку

addcount – добавлено новых записей

beforeadd – ранее добавлено (запись из списка на загрузку, которая уже существовала в TimeControl)

updatecount – количество обновленных записей (запись с ключевым полем присутствовала, но были изменены и обновлены реквизиты)

deletecount – количество удаленных записей из TimeControl (есть в TimeControl за период загрузки, но нет в списке на загрузку)

personnotfound – количество ненайденных сотрудников в TimeControl

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/absences/addfromlist

## Получения списка UID сотрудников из TimeControl для задания фото по списку (persons/needfotolist)

Получения списка сотрудников, которым можно задать\обновить фото в TimeControl.

**Название команды:**

persons/needfotolist

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    " ОрганизацияИНН": "12345678",

"ПодразделениеКод": "00001",

    " ТолькоЕслиНетФото": false,

}

Где

ОрганизацияИНН – ИНН организации (юридического лица), к которой принадлежат сотрудники. Если не заполнить, то в список попадут все сотрудники

ПодразделениеКод – текстовый код подразделения. Если указан, то будут выгружаться только отсутствия сотрудников данного подразделения

ТолькоЕслиНетФото – если true, то в список попадут только те сотрудники, у которых нет фото в TimeControl

**Результат:**

200 OK

{

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            " TABNUM": "0001",

            " ONEC\_INN": "12345678",

            " UID": "1",

            " ONEC\_ID": "0059",

            " PHOTOID": "0",

            " FACEDEEP": "0",

            " ONEC\_SIZE": "0"

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список сотрудников, для которых нужно загрузить, обновить фотографию

rec\_no – число номер записи

TABNUM – строка табельный номер сотрудника в TimeControl

ONEC\_INN – строка ИНН организации сотрудника в TimeControl (будет пустой, если сотрудники загружались в TimeControl не из 1С)

UID – внутренний идентификатор сотрудника в базе TimeControl

ONEC\_INN – строка ФизичнскоеЛицоКод сотрудника в TimeControl (будет пустой, если сотрудники загружались в TimeControl не из 1С)

PHOTOID - внутренний идентификатор фотографии сотрудника в базе TimeControl (0, если сотруднику не задано фото в TimeControl)

FACEDEEP – если сотруднику задана фотография в TimeControl, то 1 – фотография используется для устройств распознавания лиц, 0 – не используется

ONEC\_SIZE - если сотруднику задана фотография в TimeControl, то в данном поле будет размер фотографии в байтах. По нему можно определить перед загрузкой, обновлять фото или нет

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/ persons/needfotolist

## Загрузка фотографии сотрудника по UID (persons/setfoto)

Загрузка фотографии сотрудника по его UID. Список UID сотрудников, которым можно задать\обновить фото, можно получить командой [persons/needfotolist](#_Получения_списка_UID)

**Название команды:**

persons/setfoto

**Параметры:**

Отсутствуют

**Тело запроса:**

{

    " UID": 1,

    " PHOTOID": 0,

    " ONEC\_SIZE": 5555555,

    " ONLY\_FACEDEEP": false,

    " FACEDEEP": true,

    " PHOTO\_DATA": "==DFGSDFGSDFGSDFG4T34G34GRG………."

}

Где

UID – внутренний идентификатор сотрудника в базе TimeControl

PHOTOID - внутренний идентификатор фотографии сотрудника в базе TimeControl (0, если сотруднику не задано фото в TimeControl), предварительно полученный командой [persons/needfotolist](#_Получения_списка_UID)

ONEC\_SIZE – размер загружаемой фотографии в байтах

ONLY\_FACEDEEP – если true, то обновляется признак фотографии FACEDEEP (использовать для устройств распознавания лиц) без загрузки самой фотографии

FACEDEEP – если true, то фотография будет использоваться для устройств распознавания лиц, false – не будет

PHOTO\_DATA – строка данные фотографии в кодировке Base64

**Результат:**

204 OK

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/absences/addfromlist

# Экспорт данных из TimeControl (в том числе для обмена с 1С)

## Экспорт отгулов и прогулов списком (absences/getto1c)

Выгрузка отгулов и прогулов из TimeControl.

**Название команды:**

absences/getto1c

**Параметры:**

Отсутствуют

**Тело запроса:**

{

" ДатаНачала": "01.01.2023",

" ДатаОкончания": "31.01.2023",

"ОрганизацияИНН": "12345678",

"ПодразделениеКод": "0001",

"ГрузитьПрогулы": true,

"ГрузитьОтгулы": true

}

Где,

ДатаНачала и ДатаОкончания - начало и конец периода выгрузки документов отсутствий

ОрганизацияИНН – фильтр по ИНН организации, к которой принадлежит сотрудник

ПодразделениеКод – текстовый код подразделения. Если указан, то будут выгружаться только отсутствия сотрудников данного подразделения

ГрузитьПрогулы – true - выгружаются прогулы, false - не выгружаются

ГрузитьОтгулы – если true, то выгружаются отгулы, если false, то не выгружаются

**Результат:**

200 OK

{

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "ФИО": "Иванов Иван Иванович",

            "ТабельныйНомер": "00001",

            "ФизическоеЛицоКод": "059",

            "ОрганизацияИНН": "12345678",

            "ДатаДокумента": "10.01.2023",

            "ДатаНачала": "10.01.2023",

            "ДатаОкончания": "13.01.2023",

            "Номер": "0001",

            "Комментарий": "",

            "Тип": 1,

            "БуквенныйКод": "Б",

            "ЦифровойКод": "19",

            "Внутрисменны": false,

            "ЧасовВнутриСмены": 0

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список отсутствий выгруженных из TimeControl

rec\_no – число номер записи

ФИО – ФИО сотрудника

ФизическоеЛицоКод – строка код физического лица

ДатаДокумента – дата документа

ДатаНачала – начало периода отсутствия

ДатаОкончания – конец периода отсутствия

Внутрисменны – если true, то отсутствие считается внутрисменным

ЧасовВнутриСмены – количество часов для внутрисменного отсутствия

БуквенныйКод – буквенный код отсутствия

ЦифровойКод – цифровой код отсутствия

ОрганизацияИНН – ИНН организации (юридического лица), к которой принадлежит сотрудник

Тип – 4 – отгул, 5 – прогул

Номер – номер документа отсутствия

ТабельныйНомер – табельный номер сотрудника

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/absences/getto1c

## Экспорт табеля (persons/gettabel)

Выгрузка табеля из TimeControl.

**Название команды:**

persons/gettabel

**Параметры:**

Отсутствуют

**Тело запроса:**

{

" ДатаНачала": "2024-02-29T21:00:00Z",

" ДатаОкончания": "2024-03-31T20:59:59Z",

"ОрганизацияИНН": "12345678",

"ТипДанныхТабеля": 1,

"ТипОтсутствийТабеля": 0,

"ОтображатьЧасовОтсутствий": false,

"ОтображатьЧасовДляНН": false,

"РазбиватьНаСутки": false,

"ИспользоватьНочные": false,

"НочныеСтарт": "22:00",

"НочныеКонец": "07:00"

}

Где

ДатаНачала и ДатаОкончания - начало и конец периода выгрузки в форматe JSON данных табеля (начало и конец должны быть в рамках одного календарного месяца)

ОрганизацияИНН – фильтр по ИНН организации, к которой принадлежат сотрудники

ТипДанныхТабеля – 1 – по фактически отработанному времени TimeControl, 0 – плановые часы из TimeControl (график работ)

ТипОтсутствийТабеля – не используется на данный момент

ОтображатьЧасовОтсутствий – если true, то для дней отсутствий выгружается плановое количество часов

ОтображатьЧасовДляНН - если true, то для дней прогулов НН (график задан, а сотрудник по факту отсутствовал) выгружается плановое количество часов

РазбиватьНаСутки - если true, то переходящие через полночь смены разбиваются на два календарных дня

ИспользоватьНочные - если true, то будут посчитаны ночные часы с учетом интервала ночных часов с НочныеСтарт и НочныеКонец

**Результат:**

200 OK

{

    "record\_count": 1,

    "data": [

        {

            "rec\_no": 1,

            "ФИО": "Иванов Иван Иванович",

            "ТабельныйНомер": "00001",

            "ФизическоеЛицоКод": "059",

            "ОрганизацияИНН": "12345678",

            "H1": 8,

            ………………

            "H31": 0,

            "N": “Я”,

            ………………

            "N31": “”,

            "HN1": 0,

            ………………

            "HN31": 0,

            "HT1": “8”,

            ………………

            "H3T1": “0”,

            "DT": 0,

            ………………

            "DT31": 0

        }

    ]

}

Где

record\_count – количество записей в разделе data (данные)

data – список сотрудников, выгруженных из TimeControl с данными табеля на 1 месяц

rec\_no – число номер записи

ФИО – ФИО сотрудника

ФизическоеЛицоКод – строка код физического лица

ОрганизацияИНН – ИНН организации (юридического лица), к которой принадлежит сотрудник

ТабельныйНомер – табельный номер сотрудника

H1 … H31 – количество часов по дням месяца

N1 … N31 – буквенный код по дням месяца

HN1 … HN31 – количество ночных часов по дня месяца (если параметр ИспользоватьНочные был true)

DT1 … DT31 – тип дня по дням месяца (0 – обычный день, 1 – больничный, 2 – отпуск, 3 – командировка, 4 – отгул, 5 – прогул )

Если код ответа 500, или 405, то смотреть раздел [«возврат ошибки».](#_Возврат_ошибки)

**Пример вызова:**

http://127.0.0.1:5053/api/persons/gettabel

# ИСТОРИЯ ИЗМЕНЕНИЯ ВЕРСИЙ RestAPI

|  |  |  |
| --- | --- | --- |
| Дата | Версия | Изменения |
| 07.03.2024 | 1.7 | Добавлена команда revenue (журнал выручки) |
|  |  | Добавлена команда revenue/add (добавление выручки в журнал) |
|  |  | Добавлена команда revenue/delete (удаление выручки из журнала) |
| 08.11.2023 | 1.6 | В команду persons/edit добавлена возможность в качестве параметра отбора использовать DEVICE\_UID (код сотрудника на устройстве) и TABNUM (табельный номер). Должен быть указан один из параметров в порядке приоритета: UID, DEVICE\_UID, TABNUM. |
|  |  | В команду persons/getfoto добавлена возможность в качестве параметра отбора использовать DEVICE\_UID (код сотрудника на устройстве) и TABNUM (табельный номер). Должен быть указан один из параметров в порядке приоритета: UID, DEVICE\_UID, TABNUM. |
|  |  | Добавлена команда persons/setfoto\_adv (загрузить фото сотрудника из файла или данных тела запроса) |
| 12.10.2023 | 1.5 | В команду fact\_events добавлены поля TABNUM (табельный номер) |
|  |  | В команду fact\_events добавлены поля DOOR\_CODE (текстовый код проходной) |
| 25.05.2023 | 1.3 | В команду persons/getfoto добавлен параметр small |
| 02.02.2023 | 1.1 | Добавлена команда persons (справочник сотрудников) |
|  |  | Добавлена команда select (выполнение SQL запроса) |
|  |  | Добавлена команда doljnost/add (добавление должности) |
|  |  | Добавлена команда doljnost/delete (удаление должности) |
|  |  | Добавлена команда department/add (добавление подразделения) |
|  |  | Добавлена команда department/delete (удаление подразделения) |
|  |  | Добавлена команда persons/add (добавление сотрудника) |
|  |  | Добавлена команда persons/delete (удаление сотрудника) |
|  |  | Добавлена команда persons/edit (редактирование сотрудника) |
|  |  | Добавлена команда persons/getfoto (получить фото сотрудника) |
|  |  | Добавлена команда fact\_events (получить события регистрации) |
|  |  | Добавлена команда fact\_events/add (добавление события регистрации) |
|  |  | Добавлена команда time/intervals (получить интервалы присутствия) |
|  |  | Добавлена команда time/works (получить рабочее время по дням) |