

API

API `getScheduleOperation()` [Personal API](#).

```
<?php

$userToken = Intellectmoney::GetUserToken([
    //
    'login' => 'myLogin',
    'password' => 'myPassword',

    // API
    'token' => 'Bearer myToken',
    'signSecretKey' => 'mySignSecretKey',
]);

Intellectmoney::GetScheduledOperationData([
    //
    'userToken' => $userToken,
    'skip' => null,
    'take' => 10,
    'operation' => null,
    'objectId' => null,
    'objectTypeVal' => null,
    'paramsJson' => null,
    'repeatPlan' => null,
    'retryOnFailPlan' => null,
    'retryOnFailCount' => null,
    'isSingle' => null,
    'fireOnSkip' => null,
    'state' => null,
    'dateFrom' => null,
    'dateTo' => null,

    // API
    'token' => 'Bearer myToken',
    'signSecretKey' => 'signSecretKey',
]);

class IntellectMoney
{
    // API
    public static function GetUserToken($params) {
        // URL
        $url = 'https://api.intellectmoney.ru/personal/user/getUserToken';

        // API , sign (SHA256)
        $signSignature = [
            '',
            $params['login'],
            $params['password'],
            '',
            $params['signSecretKey'],
        ];
        $signHash = hash('sha256', implode(':', $signSignature));

        //
        $headers = [
            'Authorization: '.$params['token'],
            'Sign: '.$signHash,
        ];

        // ()
        $data = [
```

```

        'login' => $params['login'],
        'password' => $params['password'],
    ];

    //
    $response = Intellectmoney::CurlRequest($url, $headers, $data);
    $responseXml = simplexml_load_string($response);
    $userToken = $responseXml->Result->UserToken;

    return (string)$userToken;
}

//
public static function GetScheduledOperationData($params) {
    // URL
    $url = 'https://api.intellectmoney.ru/personal/scheduler/getscheduledoperationdata';

    // API , sign (SHA256)
    $signSignature = [
        $params['userToken'],
        $params['skip'],
        $params['take'],
        $params['operation'],
        $params['objectId'],
        $params['objectTypeVal'],
        $params['paramsJson'],
        $params['repeatPlan'],
        $params['retryOnFailPlan'],
        $params['retryOnFailCount'],
        $params['isSingle'],
        $params['fireOnSkip'],
        $params['state'],
        $params['dateFrom'],
        $params['dateTo'],
        '',
        $params['signSecretKey'],
    ];

    $signHash = hash('sha256', implode(':', $signSignature));

    //
    $headers = [
        'Authorization: '.$params['token'],
        'Sign: '.$signHash,
    ];

    // ()
    $data = [
        'userToken' => $params['userToken'],
        'skip' => $params['skip'],
        'take' => $params['take'],
        'operation' => $params['operation'],
        'objectId' => $params['objectId'],
        'objectTypeVal' => $params['objectTypeVal'],
        'paramsJson' => $params['paramsJson'],
        'repeatPlan' => $params['repeatPlan'],
        'retryOnFailPlan' => $params['retryOnFailPlan'],
        'retryOnFailCount' => $params['retryOnFailCount'],
        'isSingle' => $params['isSingle'],
        'fireOnSkip' => $params['fireOnSkip'],
        'state' => $params['state'],
        'dateFrom' => $params['dateFrom'],
        'dateTo' => $params['dateTo'],
    ];

    //
    $response = Intellectmoney::CurlRequest($url, $headers, $data);
    print_r((string)$response);
}

private static function CurlRequest($url, $headers, $data) {
    $queryString = http_build_query($data);

```

?>

IntellectMoney, .

