

SortMyBooks– API

(Application programming interface)

Welcome to Sort My Books. This documentation will help you to get started with SortMyBooks API.

General Considerations

SortMyBooks works with objects such as Company, Invoice, Product and so on. Each object is stored and identified by ID which is permanently assigned to your records.

Several objects in the system can have common properties. For example, Purchase and Invoice are very similar by their nature, although are defined as separate entities.

When accessing company through API, you need a couple of things.

- Your user id should have access to API feature. This is by default available to the owner of the company if not the owner of the company can grant you this permission.
- You will need to generate your API key. This key is used for identification when you are connecting through API. You must protect your key and if you think it's compromised, generate new one.

All the requests to the API will be automatically logged and presented under audit log.

Getting API key

Get the API Key - Login into SortMyBooks account under Home/Settings you will see tab: "API".

The screenshot shows the 'API' tab in the SortMyBooks settings. At the top, there are navigation tabs: General, Address, Accounting, Subscription, Sharing, Cloning, and API. Below the tabs, there is a section for API settings. It includes a checkbox labeled 'Enable API' which is checked. Next to it is a text input field for the 'API Secret Key' containing the value 'f09cfc852253eab8b253cb7b14718d6'. To the right of the input field is a button labeled 'Generate New Key'. Below this section is a heading 'List of API Integrations'. Under this heading, there are two integration cards. The first card is for 'PayPal' and includes the text: 'Accept Credit Card & Debit Card Payment from your customers using the PayPal Integration'. Below the text is a button labeled 'Disconnect with Paypal'. The second card is for 'Dropbox' and includes the text: 'Download your invoices, bank statements, VAT and Other Reports or upload and view your receipts and purchase invoices from Dropbox to SortMyBooks using the the Dropbox integration'. Below the text is a button labeled 'Disconnect with DropBox'. At the bottom right of the interface, there are two buttons: 'OK' and 'Cancel'.

There you will be able to generate yourself a hash key as well as enable API use for your account. Hash is a substitute for both username and password therefore must be kept in secret.

Getting Started

If you are not familiar with SOAP, then probably JSON would be easier to start. Clicking on this URL: <https://sortmybooksonline.com/api/json/company/list> will provide with a sample form with all the possibly input options

JSON RESTful access

To use this API you will only need to send HTTP POST data to a specific URL and the resulting response will be passed back in JSON format. If no POST data has been sent, you will be presented with the HTML form where you can manually specify values. This can simplify your testing.

The URL is constructed by starting with <https://sortmybooksonline.com/api/json/> and appending name of the entity you wish to work.

Calling JSON APIs

Each JSON call is determined by "type" and "operation". For example "type" may be "product" and "operation" might be "list".

Each combination of type and operation can have it's own set of arguments. The URL is formed like this:

<https://sortmybooksonline.com/api/json/product/list>

Where "product" and "list" may be substituted as needed. All the additional data is passed through POST data. If "hash" argument is not passed in POST data, the request will not be executed and the URL will return a sample HTML form with fields.

This form is a easy way to test API. You can simply fill in fields and have them sent to the API. Your remote calls should be identical to that.

A full list of types and available operations can be viewed using this link:

<https://sortmybooksonline.com/api/json/>

Hard and Soft referencing

Each object has its unique ID in the system; however you can also use soft referencing. For example you can use invoice's "Reference" number instead of ID with invoices. Similarly you can use name of the product or service.

You must be aware that while Hard references can never be changed (only deleted), soft references can be changed by user.

Standard Operations

Some operations may exist for different types, possibly with slight differences. For example, product/list is very similar to service/list. Below is the list of operations which are available for almost any type (hash means associative array):

describe	array of hashes	Describes object type returning typical list of fields, their types and captions.
list	array of hashes	Each hash describes one entry in the system. Field keys match fields as returned by "describe". Some fields may be omitted.
read	hash	Returns all record data for object specified by id or key.
update	hash	Updates record with new values. Returns all record data after it have been updated.
delete	boolean	deletes specified record. Returns true or false. Attempt to delete nonexistant record will produce error
add	hash	Creates new entry. Returns all record data after it have been added.
readall	array of hashes	Returns all record data for object specified by filter (ref_no, doc_date, is_cn). Enable only in sale and purchase types.

System selection

API can operate with multiple company data. The company is identified through field "system_id" which often is present. All the requests require "system_id" to be passed along.

Error Handling

If request cannot be processed, exception is raised. In this case, request will not return valid JSON data but will rather return an error message. Sample error message would look like this:

Exception_JsonApi_Auth: Authentication hash is invalid.

A first token can be used to identify and classify the error, the rest is human-readable error message. Below are classifications of errors starting with Exception_JsonApi_

Process	Root Cause	Fix
Auth	Authentication mechanism couldn't match your hash against any existing ones. Hash might have been changed or API might have been turned off.	Fix hash
Usage	Use of arguments is incorrect. Make sure arguments are type correctly.	Check your code
Validation	Argument values did not pass validation. Make sure that you use proper format for all the types.	Check your code
Logic	Problem with Sort My Books logic. You might be trying to edit reconciled payment or delete service which is in use. It's safe to report error description to the user.	Report to user
Permission	Insufficient permissions for operation.	not sure
Fault	Problem in SortMyBooks. Something happened what shouldn't have happened.	Report to our support

Note:-

- 1) Ensure that your company is using a valid credit card. If your credit card expires, API will stop working too.
- 2) API wouldn't also work until the sign-up wizard is completed properly.

Definition of entities

When you call "describe" command, it returns list of fields and additional information as detailed in the table below:

name	string	System-name of the field. Defined in low-case English letters with underscores.
type	one of "int", "sting", "date", "datetime", "money", "real", "numeric", "text", "boolean", "reference", "list"	Defines type of the field. Additional types may be introduced, so if field has unknown type, ignore it.
caption	string	Human-readable label for the field.
editable	boolean	Can this field be changed?

All entities will always have "id" column. ID will never change for particular record and it is numeric.

Definition of types

Type as described in previous table can have different values. This is further explained in the next table:

Type	Example in JSON	Notes
int	42	
string	"Hello World"	Maximum length is 250
date	"2010-06-20"	yyyy-mm-dd GMT
datetime	"2010-02-20 21:42:53"	yyyy-mm-dd hh-mm-ss GMT
money	29932.29	Can be negative
real	23.2889	rational number with finite decimal representation

numeric	2993	Integer. can be negative
text	"Hello World Hello World Hello World Hello.."	String with maximum length of 64,000 characters.
boolean	true	Either true or false
reference	2833	points to ID of different entity
list	"cash"	Value, one of pre-defined list such as "cash" and "invoice" for company VAT calculation basis.
json	{"is_cn":"Y"}	Valid json string

Understanding types

SortMyBooks consists of many types of entities such as invoices, purchases, payments, products, vat periods, reconciliations etc. Those types are organised hierarchically. For example there is internal type "document" which is used as a base for invoices, payments, transfers etc. Invoice type is used and extended by "sales" and "purchase" invoices.

Support

Email us your questions and suggestions on support@sortmybooks.com or better still log your ticket using <https://sortmybooks.zendesk.com/forums> and we will get back to you within 24hours with a response.

Future expansion of API

When enabling API, always make sure that user's email is specified correctly. When any major change will be introduced in API, all the users with enabled API will be notified.

Minor changes such as addition of new types, introduction of new fields or new operations may not be announced. Your software must properly ignore unrecognized data or handle it generically.

Sample Code:

Sample post call for customer add:

```
<?php
$data=array(
'hash'=>she can get it from home>settings>api key
'system_id'=>system_id,
'legal_name'=>'Test Customer',
'vat_registered'=>'Y',
'job_id'=>job_id,//https://sortmybooksonline.com/api/json/job/list
'currency_id'=>1,//https://sortmybooksonline.com/api/json/currency/list
'country_id'=>1,//https://sortmybooksonline.com/api/json/country/list
'payee_id'=>5,//1 => Supplier,2=>Employee,3=>Sub-Contractor, 4 => Other(supplier), 5 => Customer, 6 =>
Principal Contractor, 8 => Other(client)
);
/*
NB: I am only adding the mandatory fields other fields can be added as required
*/
$url='https://sortmybooksonline.com/api/json/client/add';
$ch = curl_init();
curl_setopt($ch,CURLOPT_URL,$url); curl_setopt($ch,
CURLOPT_POST,true); curl_setopt($ch,
CURLOPT_POSTFIELDS,$data);
curl_setopt($ch,CURLOPT_RETURNTRANSFER,true);
$result=curl_exec($ch); curl_close($ch);
echo "result:<pre>";
print_r($result=json_decode($result,true));
?>
```

Sample post call for invoice add:

```
/*adding invoice is a two step process
```

```
1) create the main invoice
```

```
2) adding lines to it.
```

```
*/
```

```
//adding main invoice(mandatory data)
```

```
//all the data are only sample data
```

```
//lets add sales invoice
```

```
$data_invoice_main=array(
```

```
'hash'=>'abcd1234',//get it from settings api section
```

```
'system_id'=>111,//get it from https://sortmybooksonline.com/api/json/company/list
```

```

'ref_no'=>'test invoice from API',// invoice no as you want

'contractor_to'=>9793,//get it from https://sortmybooksonline.com/api/json/client/list

'currency_id'=>1,//not mandatory, needed when creating foreign currency invoice

'exchange_rate',//not mandatory, needed when creating foreign currency invoice

'doc_date'=>'2013-07-17',//invoice date

'due_date'=>'2013-08-17',//invoice due date

'nominal_id'=>30739//should not be a parent nominal, https://sortmybooksonline.com/api/json/nominal/list

);

$url='https://sortmybooksonline.com/api/json/sale/add';

$ch = curl_init();

curl_setopt($ch, CURLOPT_URL,$url);

curl_setopt($ch,CURLOPT_POST,true); curl_setopt($ch,

CURLOPT_POSTFIELDS,$data_invoice_main); curl_setopt($ch,

CURLOPT_RETURNTRANSFER,true);

$result=curl_exec($ch); curl_close($ch);

$result=json_decode($result,true);

//adding invoice line(mandatory data)

$data_invoice_line=array(

'hash'=>'abcd1234',//get it from settings api section

'system_id'=>111,

'dochead_id'=>$result['response']',//id which is generated in main invoice

```

```
'article_id'=>1771, // get it from products or service lists
https://sortmybooksonline.com/api/json/product/list
```

```
'total_net'=>10,
```

```
'vate_rate_id'=>19,
```

```
'total_gross'=>12.3,
```

```
'total_vat'=>2.3,
```

```
'nominal_id'=>30739 // should not be a parent nominal
```

```
);
```

```
$url='https://sortmybooksonline.com/api/json/salespec/add';
```

```
$ch = curl_init();
```

```
curl_setopt($ch, CURLOPT_URL, $url);
```

```
curl_setopt($ch, CURLOPT_POST, true); curl_setopt($ch,
```

```
CURLOPT_POSTFIELDS, $data_invoice_line); curl_setopt($ch,
```

```
CURLOPT_RETURNTRANSFER, true);
```

```
$invoice_line_id=curl_exec($ch); curl_close($ch);
```

```
echo "invoice created successfully";
```