

# SharpSpring API v1.117

Major Versions: [v1](#)

[Overview](#)[Example Code](#)[Methods](#)[Schema](#)[Error Codes](#)

## Overview

SharpSpring API allows your application to connect to the internal functionality of SharpSpring. Our current implementation involves features such as:

- Creating and Updating Leads (including custom fields)
- Integrating with the Automation Engine via lead field change events (this is automatic)
- Creating, updating, and retrieving Campaigns, Accounts, and Opportunities

## Feedback

SharpSpring values the feedback of our clients. We want to provide a developer experience that allows seamless integrations, but we need your help! If you have any questions, comments, or ideas for our API, please [contact support](#).

## API Keys

In order to access the sharpspring API, you must generate both an account ID and an API key. Our API currently does not support sessions, so these keys must be supplied during each request.

Click [here](#) to generate your API keys.

## Request & Response Format

All requests are made using an HTTPs POST encoded in JSON. We use a standard very similar to JSON-RPC for request handling.

Requests must be made to the following URL:

**`https://api.sharpspring.com/pubapi/{api-version-here}`**

**v1**

**`https://api.sharpspring.com/pubapi/v1/`**

Time zone arguments passed to this endpoint will be handled based on the time zone setting selected in the Company Profile

**v1.2**

**`https://api.sharpspring.com/pubapi/v1.2/`**

Time zone arguments passed to this endpoint will be handled in UTC

Authentication parameters must be appended to the URL as query strings, for example:

**`accountID={account-id-here}&secretKey={secret-key-here}`**

Each request must contain the following three fields:

- method** - the name of the API method being called
- params** - a parameters hash to be passed to the method
- id** - a user-supplied requestID for correlating requests

Each API response contains the following three fields:

- result** - The return value of the method called. Usually a list of Objects and Object-level errors
- error** - An API-Level error returned by the API
- id** - A user-supplied request ID for correlating requests

The following is an example of using the Core::getLead function to retrieve a single lead by its ID

```
1  "Request Data:"
2
3  {
4    "method":"getLead",
5    "params":{
6      "id":"<a lead id>"
7    },
8
9    "id":"<your request ID>",
10 }
11
12 "Response Data:"
13
14 {
15   "result":{
16     "lead":[
17       {
18         "id":"<id>",
19         "companyName":"<company name>",
20         "title":"<title>",
21         "firstName":"<first name>",
22         "lastName":"<last name>",
23         "street":"<street>",
24         "city":"<city>",
25         "country":"<country>",
26         "state":"<state>",
27         "zipcode":"<zip code>",
28         "emailAddress":"<email>",
29         "website":"<website>",
30         "phoneNumber":"<phone number>",
31         "mobilePhoneNumber":"<mobile phone number>",
32         "faxNumber":"<fax number>",
33         "description":"<description>",
34         "leadScore":"<lead score>",
35         "industry":"<industry>",
36         "active":"<is active>",
37         "isUnsubscribed":"<is unsubscribed>",
38         "leadStatus":"<lead status>"
39       }
40     ]
41   },
42   "error":null,
43   "id":"<your request ID>"
44 }
```

gistfile1.json hosted with ❤️ by GitHub

[view raw](#)

For more information about methods and parameters, please view the [methods](#) section of the API documentation.

For example code, please view the [examples](#) section of the API documentation.

## API-Level Errors

When an invalid request is made to the API, an error object will be returned. Here is an example API-level error object:

```

1  {
2    "result":null,
3    "error":{
4      "code":205,
5      "message":"Invalid parameters",
6      "data":{
7        "params":[
8          {
9            "param":"id",
10           "message":"Expected data of type integer",
11           "validFormat":{
12             "type":"int",
13             "length":11,
14             "required":false
15           }
16         ]
17       }
18     },
19   },
20   "id":"<request ID>"
21 }

```

gistfile1.json hosted with ❤️ by GitHub

[view raw](#)

## Object-Level Errors

When an invalid-format object is passed as part of a list (for example, in a create or update query), an object-level error will be returned in the result array. Since the error was at the object level, no API errors will be returned. Here is an example response with an Object-level error object:

```

1  {
2    "result":{
3      "creates":[
4        {
5          "success":"false",
6          "error":{
7            "code":301,
8            "message":"Entry already exists",
9            "data":[]
10         }
11       ]
12     },
13   },
14   "error":null,
15   "id":"<request ID>"
16 }

```

gistfile1.json hosted with ❤️ by GitHub

[view raw](#)

## API Usage Limitations

Each SharpSpring partner is currently allotted 5,000 API Requests per day by default with a maximum of 500 queries per request. Requests may be rate limited in times of high traffic. The amount of API requests is subject to change in the future.

# SharpSpring API v1.117

Major Versions: [v1](#)

[Overview](#)[Example Code](#)[Methods](#)[Schema](#)[Error Codes](#)

## Using PHP & cURL

A common method of connecting to APIs is using the low level cURL utility. The following example demonstrates a simple request to the SharpSpring API using PHP. Notice the content type for the request is manually set to JSON and data is encoded in JSON format.

```
1  <?php
2      /** Get all leads with a limit of 500 results */
3      $limit = 500;
4      $offset = 0;
5
6      $method = 'getLeads';
7      $params = array('where' => array(), 'limit' => $limit, 'offset' => $offset);
8      $requestID = session_id();
9      $accountID = '<account-id>';
10     $secretKey = '<secret-key>';
11     $data = array(
12         'method' => $method,
13         'params' => $params,
14         'id' => $requestID,
15     );
16
17     $queryString = http_build_query(array('accountID' => $accountID, 'secretKey' => $secretKey));
18     $url = "http://api.sharpspring.com/pubapi/v1/?$queryString";
19
20     $data = json_encode($data);
21     $ch = curl_init($url);
22     curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "POST");
23     curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
24     curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
25     curl_setopt($ch, CURLOPT_HTTPHEADER, array(
26         'Content-Type: application/json',
27         'Content-Length: ' . strlen($data),
28         'Expect: '
29     ));
30
31     $result = curl_exec($ch);
32     curl_close($ch);
33
34     echo $result;
35
36  ?>
```

gistfile1.php hosted with ❤ by GitHub

[view raw](#)

## JSON structure for createLeads

Basic example of how to use the `createLeads` method

```
1 {
2   "method":"createLeads",
3   "params":
4     {"objects":
5       [
6         {"firstName":"foo","lastName":"bar","emailAddress":"foo@bar.com"},
7         {"firstName":"fizz","lastName":"buzz","emailAddress":"fizz@buzz.com"}
8       ]
9     },
10   "id":"<your request ID>"
11 }
12 }
```

gistfile1.json hosted with ❤ by GitHub

[view raw](#)

## JSON structure for updateLeads

Basic example of how to use the **updateLeads** method

```
1 {
2   "method":"updateLeads",
3   "params":
4     {"objects":
5       [
6         {"id":"<lead ID>","firstName":"fooUpdate","lastName":"barUpdate"}
7       ]
8     },
9   "id":"<your request ID>"
10 }
```

gistfile1.json hosted with ❤ by GitHub

[view raw](#)

## JSON structure for deleteLeads

Basic example of how to use the **deleteLeads** method

```
1 {
2   "method":"deleteLeads",
3   "params":
4     {"objects":
5       [
6         {"id":"<lead ID>"},
7         {"id":"<another lead ID>"},
8       ]
9     },
10   "id":"<your request ID>"
11 }
```

gistfile1.json hosted with ❤ by GitHub

[view raw](#)

# SharpSpring APIv1.117

Major Versions: [v1](#)

Overview

Example Code

Methods

Schema

Error Codes

Method	Module	Table
<a href="#">createAccounts</a>	Core	Account
<a href="#">deleteAccounts</a>	Core	Account
<a href="#">getAccount</a>	Core	Account
<a href="#">getAccounts</a>	Core	Account
<a href="#">getAccountsDateRange</a>	Core	Account
<a href="#">updateAccounts</a>	Core	Account
<a href="#">createCampaigns</a>	Core	Campaign
<a href="#">deleteCampaigns</a>	Core	Campaign
<a href="#">getCampaign</a>	Core	Campaign
<a href="#">getCampaigns</a>	Core	Campaign
<a href="#">getCampaignsDateRange</a>	Core	Campaign
<a href="#">updateCampaigns</a>	Core	Campaign
<a href="#">getClients</a>	Core	Company Profile Managed By
<a href="#">createSendMail</a>	Core	CreateSendEmail
<a href="#">createDealStages</a>	Core	DealStage
<a href="#">deleteDealStages</a>	Core	DealStage
<a href="#">getDealStage</a>	Core	DealStage
<a href="#">getDealStages</a>	Core	DealStage
<a href="#">getDealStagesDateRange</a>	Core	DealStage
<a href="#">createEmail</a>	Core	Email
<a href="#">deleteEmail</a>	Core	Email
<a href="#">getEmail</a>	Core	Email
<a href="#">getEmailListing</a>	Core	Email
<a href="#">updateEmail</a>	Core	Email
<a href="#">deleteScheduleSend</a>	Core	EmailJob
<a href="#">getEmailEvents</a>	Core	EmailJob
<a href="#">getEmailSendEvents</a>	Core	EmailJob
<a href="#">getEmailSendStatus</a>	Core	EmailJob
<a href="#">getScheduleSendStatus</a>	Core	EmailJob

getScheduleSends	Core	EmailJob
getSendCount	Core	EmailJob
getEvents	Core	Event
createFields	Core	Field
deleteFields	Core	Field
getFields	Core	Field
updateFields	Core	Field
getRelatedLeadsByEmail	Core	Form Event History
getListTags	Core	GetListTags
createLeads	Core	Lead
deleteLeads	Core	Lead
getLead	Core	Lead
getLeads	Core	Lead
getLeadsDateRange	Core	Lead
subscribeToLeadUpdates	Core	Lead
updateLeads	Core	Lead
updateLeadsV2	Core	Lead
createList	Core	List
deleteList	Core	List
getActiveLists	Core	List
updateList	Core	List
addListMember	Core	ListMember
addListMemberEmailAddress	Core	ListMember
addListMembers	Core	ListMember
addListMembersEmailAddress	Core	ListMember
getContactListMemberships	Core	ListMember
getEmailSend	Core	ListMember
getListMemberships	Core	ListMember
removeListMember	Core	ListMember
removeListMemberEmailAddress	Core	ListMember
removeListMembers	Core	ListMember
removeListMembersEmailAddress	Core	ListMember
removeListMembersAll	Core	ListMembers
createOpportunities	Core	Opportunity
deleteOpportunities	Core	Opportunity

getOpportunities	Core	Opportunity
getOpportunitiesDateRange	Core	Opportunity
getOpportunity	Core	Opportunity
updateOpportunities	Core	Opportunity
createOpportunityLeads	Core	OpportunityLead
deleteOpportunityLeads	Core	OpportunityLead
getOpportunityLead	Core	OpportunityLead
getOpportunityLeads	Core	OpportunityLead
getOpportunityLeadsDateRange	Core	OpportunityLead
getUnsubscribeCategories	Core	Unsubscribe Category
getUserProfiles	Core	UserProfile
updateAccountSendLimit	Core	accountSendLimit
createClient	Core	companyProfile
updateClient	Core	companyProfileUpdate
disableAccountSending	Core	disableAccountSending
enableAccountSending	Core	enableAccountSending
logCalls	Core	eventHistory
getEmailJobs	Core	getemailjob
getLeadListMemberships	Core	leadListMembership
getLeadTimeline	Core	leadTimeline
getRemovedListMembers	Core	listLeadMember
getListMembers	Core	listMembers

Account

createAccounts

(Account)

Specify a list of Account objects to be created in SharpSpring

Params

Name	Type	Is Required
objects	list<account>	Required

Return Data

Name	Type
creates	list<createResult>

deleteAccounts

(Account)

Specify a list of Account IDs to be deleted in SharpSpring



Params

Name	Type	Is Required
objects	list<id>	Required

Return Data

Name	Type
deletes	list<deleteResult>

getAccount

(Account)

Retrieve a single Account by its ID

Params

Name	Type	Is Required
id	int	Required

Return Data

Name	Type
accounts	list<Account>

getAccounts

(Account)

Retrieve a list of Accounts given a WHERE clause, or retrieve all Accounts if WHERE clause is empty

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id,ownerID	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
accounts	list<Account>

getAccountsDateRange

(Account)

Retrieve a list of Accounts that have been either created or updated between two timestamps. Timestamps must be specified in Y-m-d H:i:s format

Params

Name	Type	Is Required
startDate	timestamp	Required
endDate	timestamp	Required

timestamp	<b>string</b> <i>Required:</i> create,update <i>Optional:</i> none	Required
-----------	--	----------

Return Data

Name	Type
Accounts	list<Account>

updateAccounts

(Account)

Specify a list of Account objects to be updated in SharpSpring

Params

Name	Type	Is Required
objects	list<account>	Required

Return Data

Name	Type
updates	list<updateResult>

Campaign

createCampaigns

(Campaign)

Specify a list of Campaign objects to be created in SharpSpring

Params

Name	Type	Is Required
objects	list<campaign>	Required

Return Data

Name	Type
creates	list<createResult>

deleteCampaigns

(Campaign)

Specify a list of Campaign IDs to be deleted in SharpSpring

Params

Name	Type	Is Required
objects	list<id>	Required

Return Data

Name	Type
deletes	list<deleteResult>

## getCampaign

(Campaign)

Retrieve a single Campaign by its ID

### Params

Name	Type	Is Required
id	int	Required

### Return Data

Name	Type
campaigns	list<Campaign>

## getCampaigns

(Campaign)

Retrieve a list of Campaigns given a WHERE clause, or retrieve all Campaigns if WHERE clause is empty

### Params

Name	Type	Is Required
where	<b>hash</b> <i>Required:</i> none <i>Optional:</i> id,campaignName	Required
limit	int	Optional
offset	int	Optional

### Return Data

Name	Type
campaigns	list<Campaign>

## getCampaignsDateRange

(Campaign)

Retrieve a list of Campaigns that have been either created or updated between two timestamps. Timestamps must be specified in Y-m-d H:i:s format

### Params

Name	Type	Is Required
startDate	timestamp	Required
endDate	timestamp	Required
timestamp	<b>string</b> <i>Required:</i> create,update <i>Optional:</i> none	Required

### Return Data

Name	Type
Campaigns	list<Campaign>

## updateCampaigns

(Campaign)

Specify a list of Campaign objects to be updated in SharpSpring

Params

Name	Type	Is Required
objects	list<campaign>	Required

Return Data

Name	Type
updates	list<updateResult>

Company Profile Managed By

getClient

(Company Profile Managed By)

Get a list of all active companies managed by your company

Params

Name	Type	Is Required

Return Data

Name	Type
getAllCompanyProfileManagedBy	list<Client>

CreateSendEmail

createSendMail

(CreateSendEmail)

Send an email to a list or lists of a specific tag.

Params

Name	Type	Is Required
objects	list<ScheduleSend>	Required

Return Data

Name	Type
actions	list<CreateResult>

DealStage

createDealStages

(DealStage)

Specify a list of DealStage objects to be created in SharpSpring

Params

--

Name	Type	Is Required
objects	list<dealStage>	Required

Return Data

Name	Type
creates	list<createResult>

deleteDealStages

(DealStage)

Specify a list of DealStage IDs to be deleted in SharpSpring

Params

Name	Type	Is Required
objects	list<id>	Required

Return Data

Name	Type
deletes	list<deleteResult>

getDealStage

(DealStage)

Retrieve a single DealStage by its ID

Params

Name	Type	Is Required
id	int	Required

Return Data

Name	Type
dealStages	list<DealStage>

getDealStages

(DealStage)

Retrieve a list of DealStage objects given a WHERE clause, or retrieve all DealStage objects if WHERE clause is empty

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
dealStages	list<DealStage>

## getDealStagesDateRange

(DealStage)

Retrieve a list of DealStages that have been either created or updated between two timestamps. Timestamps must be specified in Y-m-d H:i:s format

### Params

Name	Type	Is Required
startDate	timestamp	Required
endDate	timestamp	Required
timestamp	string Required: create,update Optional: none	Required

### Return Data

Name	Type
DealStages	list<DealStage>

## Email

### createEmail

(Email)

Create an email from html

### Params

Name	Type	Is Required
objects	list<Email>	Required

### Return Data

Name	Type
actions	list<CreateResult>

### deleteEmail

(Email)

Specify a email ID to be deleted in SharpSpring

### Params

Name	Type	Is Required
objects	list<id>	Required

### Return Data

Name	Type
actions	list<DeleteResult>

### getEmail

(Email)

Returns the requested emails html data.

Params

Name	Type	Is Required
id	int	Required

Return Data

Name	Type
fields	Returns just the html data

getEmailListing

(Email)

Returns a list of email information.

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id	Optional
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
fields	list<emailListing>

updateEmail

(Email)

Specify a list of existing Emails to be updated in SharpSpring. The 'id' of each email must be passed for each record being updated

Params

Name	Type	Is Required
objects	list<Email>	Required

Return Data

Name	Type
actions	list<UpdateResult>

EmailJob

deleteScheduleSend

(EmailJob)

Delete an email job scheduled to send to a list or a lead.

Params

Name	Type	Is Required
------	------	-------------

jobID	int	Required
-------	-----	----------

Return Data

Name	Type
deletes	list<DeleteResult>

getEmailEvents

(EmailJob)

Retrieve email statistics by jobID. Optionally filter to event type. Allowed values for eventType are {clicks, opens, bounces, sends, unsubscribes}

Params

Name	Type	Is Required
emailJobID	int	Required
eventType	string	Required

Return Data

Name	Type
events	list<EmailEvent>

getEmailSendEvents

(EmailJob)

Retrieve email sends for a given email job.

Params

Name	Type	Is Required
sendID	int	Required
eventType	string [sends,opens,clicks,bounces,spam,unsubscribes]	Required
limit	int	Required
startingAfter	int [eventID]	Optional

Return Data

Name	Type
events	list<EmailSendEvent>
hasMore	boolean

getEmailSendStatus

(EmailJob)

Retrieve a status determining whether an email job has been sent or is in queue.

Params

Name	Type	Is Required
sendID	int	Required

Return Data

Name	Type
------	------



status	n/a
--------	-----

getScheduleSendStatus

(EmailJob)

Retrieve the send status of a scheduled email job.

Params

Name	Type	Is Required
jobID	int	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
automationEventQueuee	id, taskID, workflowID, timeProcessed, processed, workflowEventData, sendDuplicate, email{id, companyID, fromEmail, fromName, subject, title}, whoID, whoType, whatID, whatType, createTimestamp

getScheduleSends

(EmailJob)

Retrieve email jobs scheduled to send by list, lead, email, or job.

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id,campaignName	Required

Return Data

Name	Type
sends	list<ScheduleSendQueue>

getSendCount

(EmailJob)

Retrieve email send counts by date range and companyID for Agency and Client Accounts

Params

Name	Type	Is Required
fromDate	timestamp	Required
toDate	timestamp	Required
companyID	int	Optional

Return Data

Name	Type
sendCounts	list<companyID,sendCount>

Event

getEvents

(Event)

Retrieve a list of SharpSpring Lead events. Either a lead ID or an event timestamp can be specified. In the case when an event timestamp is specified, the API will return all events that happened after the given timestamp.

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> leadID,createTimestamp	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
fields	list<Event>

Field

createFields

(Field)

Specify a list of field objects to be created in SharpSpring. Each field must, at minimum, have a relationship (ex. opportunity, lead, or account), dataType (text, picklist, etc.), and label.

Params

Name	Type	Is Required
objects	list<Field>	Required

Return Data

Name	Type
creates	list<CreateResult>

deleteFields

(Field)

Specify a list of fields to be deleted in SharpSpring. Only custom fields can be deleted. These fields are deleted by systemName.

Params

Name	Type	Is Required
objects	list<systemName>	Required

Return Data

Name	Type
------	------

deletes	list<deleteResult>
---------	--------------------

getFields

(Field)

Retrieve a list of Field objects. This function is extremely useful for retrieving a list of all custom fields and system fields available in SharpSpring. Every result will have a "systemName" which is the key that must be specified in order to update that field for a specific lead.

Params

Name	Type	Is Required
where	<b>hash</b> <i>Required:</i> none <i>Optional:</i> id,label,systemName,isCustom	Required
limit	<b>int</b>	Optional
offset	<b>int</b>	Optional

Return Data

Name	Type
fields	list<Field>

updateFields

(Field)

Specify a list of existing Fields to be updated in SharpSpring. The 'id' of each field must be passed for each record being updated.

Params

Name	Type	Is Required
objects	list<Field>	Required

Return Data

Name	Type
updates	list<updateResult>

Form Event History

getRelatedLeadsByEmail

(Form Event History)

Retrieve an ordered set of Form Event History records via an email. Useful for de-duping leads in your own CRM.

Params

Name	Type	Is Required
where	<b>hash</b> <i>Required:</i> none <i>Optional:</i> id,trackingID	Required
limit	<b>int</b>	Optional
offset	<b>int</b>	Optional

Return Data

Name	Type
fields	list<FormEventHistory>

GetListTags

getListTags

(GetListTags)

Returns a tag list for lists.

Params

Name	Type	Is Required
none		Optional

Return Data

Name	Type
result	list<listTag>

Lead

createLeads

(Lead)

Specify a list of Lead objects to be created in SharpSpring. Every lead object is a hash keyed by the system name of the lead field. If you wish to push custom fields, first use the "getFields" API method in order to retrieve a list of custom fields. In order to set a custom field for a lead, use the field's "systemName" attribute as the key. Note: This method accepts up to 500 lead objects per call. However, in instances with a lot of custom field data being passed, it's advisable to chunk the requests into smaller, more manageable calls to improve performance. 250 lead objects is recommended in those cases.

Params

Name	Type	Is Required
objects	list<lead>	Required

Return Data

Name	Type
creates	list<createResult>

deleteLeads

(Lead)

Specify a list of leads to be deleted in SharpSpring by ID

Params

Name	Type	Is Required
objects	list<id>	Required

Return Data

Name	Type
------	------

deletes	list<deleteResult>
---------	--------------------

getLead

(Lead)

Retrieve a single Lead by its ID

Params

Name	Type	Is Required
id	int	Required

Return Data

Name	Type
leads	list<Lead>

getLeads

(Lead)

Retrieve a list of Leads given a WHERE clause, or retrieve all Leads if WHERE clause is empty. If FIELDS is specified, only the fields requested will be returned

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id,emailAddress	Required
limit	int	Optional
offset	int	Optional
fields	list<systemName>	Optional

Return Data

Name	Type
leads	list<Lead>

getLeadsDateRange

(Lead)

Retrieve a list of Leads that have been either created or updated between two timestamps. Timestamps must be specified in Y-m-d H:i:s format. If FIELDS is specified, only the fields requested will be returned

Params

Name	Type	Is Required
startDate	timestamp	Required
endDate	timestamp	Required
timestamp	string <i>Required:</i> create,update <i>Optional:</i> none	Required
fields	list<systemName>	Optional

Return Data

--

Name	Type
leads	list<Lead>

subscribeToLeadUpdates

(Lead)

Subscribe a URL to receive real-time lead updates. We will POST a list of updated leads whenever leads are updated in SharpSpring.

Params

Name	Type	Is Required
url	string	Required

Return Data

Name	Type
success	boolean

updateLeads

(Lead)

Specify a list of Lead objects to be updated in SharpSpring. Every lead object is a hash keyed by the system name of the lead field. If you wish to push custom fields, first use the "getFields" API method in order to retrieve a list of custom fields.In order to set a custom field for a lead, use the field's "systemName" attribute as the key.

Params

Name	Type	Is Required
objects	list<lead>	Required

Return Data

Name	Type
updates	list<updateResult>

updateLeadsV2

(Lead)

Specify a list of Lead objects to be updated in SharpSpring. Every lead object is a hash keyed by the system name of the lead field. If you wish to push custom fields, first use the "getFields" API method in order to retrieve a list of custom fields.In order to set a custom field for a lead, use the field's "systemName" attribute as the key. This differs from updateLeads in that it returns error code 404 if the lead id does not exist or the lead id is not provided and the lead email does not exist.

Params

Name	Type	Is Required
objects	list<lead>	Required

Return Data

Name	Type
updates	list<updateResult>

createList

(List)

Create a static mailing list.

Params

Name	Type	Is Required
name	string	Required
description	string	Optional

Return Data

Name	Type
status	boolean<true/false>
id	int

deleteList

(List)

Specify a list of Lists to be deleted in SharpSpring by ID

Params

Name	Type	Is Required
objects	list<id>	Required

Return Data

Name	Type
deletes	list<deleteResult>

getActiveLists

(List)

Returns a list of active Lists

Params

Name	Type	Is Required
where	hash <i>Required: none</i> <i>Optional: id</i>	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
activeList	list<List>

updateList

(List)

Updates an existing list.

Params

--

Name	Type	Is Required
id	int	Required
name	string	Optional
description	string	Optional

Return Data

Name	Type
updates	list<UpdateResult>

ListMember

addListMember

(ListMember)

Adds a lead to a list

Params

Name	Type	Is Required
listID	int	Required
memberID	int	Required

Return Data

Name	Type
creates	list<createResult>

addListMemberEmailAddress

(ListMember)

Add a member to a mailing list using their email address.

Params

Name	Type	Is Required
listID	int	Required
emailAddress	string	Required

Return Data

Name	Type
create	list<CreateResult>

addListMembers

(ListMember)

Specify a list of leads by id and lists for managing memberships in bulk. Each call can handle up to 500 associations.

Params

Name	Type	Is Required
------	------	-------------



objects	list<listID, memberID>	Required
---------	------------------------	----------

Return Data

Name	Type
creates	list<createResult>

addListMembersEmailAddress

(ListMember)

Specify a list of leads by email address and lists for managing memberships in bulk. Each call can handle up to 500 associations.

Params

Name	Type	Is Required
objects	list<listID, emailAddress>	Required

Return Data

Name	Type
creates	list<createResult>

getContactListMemberships

(ListMember)

Get the lists that a particular contact is a member of

Params

Name	Type	Is Required
where	hash <i>Required:</i> contactID <i>Optional:</i> none	Required

Return Data

Name	Type
listMember	list<ListMember>

getEmailSend

(ListMember)

Retrieve all aggregate data for an email job.

Params

Name	Type	Is Required
where	hash <i>Required:</i> sendID <i>Optional:</i> none	Required

Return Data

Name	Type
EmailSend	list<EmailSend>

getListMemberships

(ListMember)

Get the lists that a particular contact is a member of by email address.

Params

Name	Type	Is Required
emailAddress	string	Required

Return Data

Name	Type
ListMembership	list<ListMembership>

removeListMember

(ListMember)

Removes a lead from a list

Params

Name	Type	Is Required
listID	int	Required
contactID	int	Required

Return Data

Name	Type
updates	list<updateResult>

removeListMemberEmailAddress

(ListMember)

Remove a member from a mailing list using their email address.

Params

Name	Type	Is Required
listID	int	Required
emailAddress	string	Required

Return Data

Name	Type
deletes	list<deleteResult>

removeListMembers

(ListMember)

Specify a list of leads by id and lists for managing memberships in bulk. Each call can handle up to 500 updates.

Params

Name	Type	Is Required
objects	list<listID, memberID>	Required

Return Data

Name	Type
------	------

updates	list<updateResult>
---------	--------------------

## removeListMembersEmailAddress

(ListMember)

Specify a list of leads by email address and lists for managing memberships in bulk. Each call can handle up to 500 updates.

### Params

Name	Type	Is Required
objects	list<listID, emailAddress>	Required

### Return Data

Name	Type
updates	list<updateResult>

## ListMembers

## removeListMembersAll

(ListMembers)

Removes all members from a static list there were not manually added. Code is the HTTP code representing the status of the request and the Message is the accompanying description of the code

### Params

Name	Type	Is Required
id	int	Required

### Return Data

Name	Type
result	code, message

## Opportunity

## createOpportunities

(Opportunity)

Specify a list of Opportunity objects to be created in SharpSpring.

### Params

Name	Type	Is Required
objects	list<opportunity>	Required

### Return Data

Name	Type
creates	list<createResult>

## deleteOpportunities

(Opportunity)

Specify a list of Opportunity IDs to be deleted in SharpSpring

### Params

Name	Type	Is Required
objects	list<id>	Required

### Return Data

Name	Type
deletes	list<deleteResult>

## getOpportunities

(Opportunity)

Retrieve a list of Opportunities given a WHERE clause, or retrieve all Opportunities if WHERE clause is empty

### Params

Name	Type	Is Required
where	<b>hash</b> <i>Required:</i> none <i>Optional:</i> id,ownerID,dealStageID,accountID,campaignID	Required
limit	int	Optional
offset	int	Optional

### Return Data

Name	Type
opportunity	list<Opportunity>

## getOpportunitiesDateRange

(Opportunity)

Retrieve a list of Opportunities that have been either created or updated between two timestamps. Timestamps must be specified in Y-m-d H:i:s format

### Params

Name	Type	Is Required
startDate	timestamp	Required
endDate	timestamp	Required
timestamp	<b>string</b> <i>Required:</i> create,update <i>Optional:</i> none	Required

### Return Data

Name	Type
opportunity	list<Opportunity>

## getOpportunity

(Opportunity)

Retrieve a single Opportunity by its ID

Params

Name	Type	Is Required
id	int	Required

Return Data

Name	Type
opportunities	list<Opportunity>

updateOpportunities

(Opportunity)

Specify a list of Opportunity objects to be updated in SharpSpring.

Params

Name	Type	Is Required
objects	list<opportunity>	Required

Return Data

Name	Type
updates	list<updateResult>

OpportunityLead

createOpportunityLeads

(OpportunityLead)

Specify a list of OpportunityLead objects to be created in SharpSpring

Params

Name	Type	Is Required
objects	list<OpportunityLead>	Required

Return Data

Name	Type
creates	list<createResult>

deleteOpportunityLeads

(OpportunityLead)

Specify a list of OpportunityLead IDs to be deleted in SharpSpring

Params

Name	Type	Is Required
objects	list<id>	Required

Return Data

Name	Type
------	------

deletes	list<deleteResult>
---------	--------------------

getOpportunityLead

(OpportunityLead)

Retrieve a single OpportunityLead by its ID

Params

Name	Type	Is Required
id	int	Required

Return Data

Name	Type
opportunityLeads	list<OpportunityLead>

getOpportunityLeads

(OpportunityLead)

Retrieve a list of OpportunityLead objects given a WHERE clause, or retrieve all OpportunityLead objects if WHERE clause is empty

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id,opportunityID,leadID	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
opportunityLeads	list<OpportunityLead>

getOpportunityLeadsDateRange

(OpportunityLead)

Retrieve a list of OpportunityLeads that have been either created or updated between two timestamps. Timestamps must be specified in Y-m-d H:i:s format

Params

Name	Type	Is Required
startDate	timestamp	Required
endDate	timestamp	Required
timestamp	string <i>Required:</i> create,update <i>Optional:</i> none	Required

Return Data

Name	Type
OpportunityLeads	list<OpportunityLead>

Unsubscribe Category

getUnsubscribeCategories

(Unsubscribe Category)

Get a list of un-subscribe categories.

Params

Name	Type	Is Required
none		Optional

Return Data

Name	Type
getAllUnsubscribeCategories	list<UnsubscribeCategory>

UserProfile

getUserProfiles

(UserProfile)

Retrieve a list of UserProfile objects

Params

Name	Type	Is Required
where	hash <i>Required:</i> none <i>Optional:</i> id,emailAddresses	Optional
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
userProfiles	list<UserProfile>

accountSendLimit

updateAccountSendLimit

(accountSendLimit)

Updates the send limit of a sub account.

Params

Name	Type	Is Required
clientID	int	Required

emailLimit	int	Required
------------	-----	----------

Return Data

Name	Type
updates	list<UpdateResult>

companyProfile

createClient

(companyProfile)

Creates a sub client for agencies.

**country:** This values is based on [ISO 3166-1 alpha-2](#) standard, example: US, UK, IT ...

**locale (optional):** is responsible for your language and needs to be one of the following accepted: *en\_US, de\_DE, es\_ES, fr\_FR, it\_IT, nl\_NL, pt\_PT, ro\_RO, sk\_SK, tr\_TR*.

**salesStartHour & salesEndHour:** These are based on a 24 hour clock, example: 00h to 24h.

Adding in a username and password will create a new user for that profile.

Params

Name	Type	Is Required
companyName	string	Required
streetAddress	string	Optional
zipCode	string	Optional
country	string	Optional
state	string	Optional
city	string	Optional
salesPhone	string	Required
defaultCurrency	string	Optional
locale	string	Optional
salesStartHour	int	Optional
salesEndHour	int	Optional
salesOfficialTimeZone	string	Optional
disallowClientRebrand	boolean	Optional
hideGoogleAdWords	boolean	Optional
displayCallTrackingSettings	boolean	Optional
userEmail	string	Optional
userPassword	string	Optional
userDisplayName	string	Optional

Return Data



Name	Type
status	boolean
error	array
userID	int
id	int

companyProfileUpdate

updateClient

(companyProfileUpdate)

Updates a sub client for agencies.

country: This values is based on [ISO 3166-1 alpha-2](#) standard, example: US, UK, IT ...

locale (optional): is responsible for your language and needs to be one of the following accepted: *en\_US, de\_DE, es\_ES, fr\_FR, it\_IT, nl\_NL, pt\_PT, ro\_RO, sk\_SK, tr\_TR*.

salesStartHour & salesEndHour: These are based on a 24 hour clock, example: 00h to 24h.

Params

Name	Type	Is Required
id	int	Required
streetAddress	string	Optional
zipCode	string	Optional
country	string	Optional
state	string	Optional
city	string	Optional
salesPhone	string	Optional
defaultCurrency	string	Optional
locale	string	Optional
salesStartHour	int	Optional
salesEndHour	int	Optional
salesOfficialTimeZone	string	Optional
disallowClientRebrand	boolean	Optional
hideGoogleAdWords	boolean	Optional
displayCallTrackingSettings	boolean	Optional

Return Data

Name	Type
updates	list<UpdateResult>

# disableAccountSending

## disableAccountSending

(disableAccountSending)

Disables a subaccount's sending.

### Params

Name	Type	Is Required
clientID	int	Required

### Return Data

Name	Type
updates	list<UpdateResult>

# enableAccountSending

## enableAccountSending

(enableAccountSending)

Enables a subaccount's sending.

### Params

Name	Type	Is Required
clientID	int	Required

### Return Data

Name	Type
updates	list<UpdateResult>

# eventHistory

## logCalls

(eventHistory)

Log calls with leads. The timestamp must be specified in Y-m-d H:i:s format

### Params

Name	Type	Is Required
userID	int	Required
leadID	int	Required
direction	string<incoming/outgoing>	Required
callResult	string<answered/noAnswerMessage/noAnswerNoMessage>	Required

callNotes	string	Optional
createTimestamp	string	Optional

Return Data

Name	Type
success	boolean

getemailjob

getEmailJobs

(getemailjob)

Get a list of active email jobs for the current account.

Params

Name	Type	Is Required
limit	string	Optional
offset	string	Optional

Return Data

Name	Type
getAllgetEmailJobs	list<EmailJob>

leadListMembership

getLeadListMemberships

(leadListMembership)

Gets the list id's for each member Id. A max of 100 leads per request

Params

Name	Type	Is Required
memberIDs	list<MemberID>	Required

Return Data

Name	Type
getWhereIns	list<ListID>

leadTimeline

getLeadTimeline

(leadTimeline)

Gets lead event history. Can be filtered by providing constraints.

Possible values for eventSource parameter: emailOpened, emailVisit, formSubmit, importantPageVisit.

Params

Name	Type	Is Required
where	hash <i>Required:</i> whoID <i>Optional:</i> eventName,eventSource,whatType,createTimestamp,whoID	Required

Return Data

Name	Type
leadTimeline	list<Event>

listLeadMember

getRemovedListMembers(listLeadMember)

Get the members of a list that are "removed", "unsubscribed" or "hardbounced". If none of these are supplied via the flag parameter then it will default to "removed".

Params

Name	Type	Is Required
where	hash <i>Required:</i> id <i>Optional:</i> flag	Required
limit	int	Optional
offset	int	Optional

Return Data

Name	Type
getWhereListLeadMembers	list<RemovedListMember>

listMembers

getListMembers(listMembers)

Returns the active members for a specific list

Params

Name	Type	Is Required
where	hash <i>Required:</i> id <i>Optional:</i> none	Required
limit	int	Optional

offset	int	Optional
--------	-----	----------

Return Data

Name	Type
fields	list<List>

---

# SharpSpring APIv1.117

Major Versions: [v1](#)

Overview	Example Code	Methods	Schema	Error Codes
----------	--------------	---------	--------	-------------

## Schema

Table	Module
<a href="#">Account</a>	Core
<a href="#">ActiveListMember</a>	Core
<a href="#">Campaign</a>	Core
<a href="#">Client</a>	Core
<a href="#">CreateResult</a>	Core
<a href="#">DealStage</a>	Core
<a href="#">DeleteResult</a>	Core
<a href="#">Email</a>	Core
<a href="#">EmailEvent</a>	Core
<a href="#">EmailJob</a>	Core
<a href="#">EmailListing</a>	Core
<a href="#">EmailSend</a>	Core
<a href="#">EmailSendEvent</a>	Core
<a href="#">Event</a>	Core
<a href="#">Field</a>	Core
<a href="#">FormEventHistory</a>	Core
<a href="#">Lead</a>	Core
<a href="#">List</a>	Core
<a href="#">ListMember</a>	Core
<a href="#">ListMembership</a>	Core
<a href="#">ListTag</a>	Core
<a href="#">Opportunity</a>	Core
<a href="#">OpportunityLead</a>	Core
<a href="#">RemovedListMember</a>	Core
<a href="#">ScheduleSend</a>	Core
<a href="#">ScheduleSendQueue</a>	Core
<a href="#">UnsubscribeCategory</a>	Core

UpdateResult	Core
UserProfile	Core

# Core Module

Core SharpSpring CRM API Functionality

## Account

An Account is an organization or person that is associated with an Opportunity.

Name	Type	Length	Is Required
id	int	18	Optional
ownerID	int	18	Optional
accountName	varchar	255	Required
industry	varchar	255	Optional
phone	varchar	255	Optional
annualRevenue	int	18	Optional
numberOfEmployees	int	18	Optional
website	varchar	255	Optional
yearStarted	int	18	Optional
fax	varchar	255	Optional
billingCity	varchar	255	Optional
billingCountry	varchar	255	Optional
billingPostalCode	varchar	50	Optional
billingState	varchar	255	Optional
billingStreetAddress	varchar	255	Optional
shippingCity	varchar	255	Optional
shippingCountry	varchar	255	Optional
shippingPostalCode	varchar	50	Optional
shippingState	varchar	255	Optional
shippingStreetAddress	varchar	255	Optional

## ActiveListMember

The ActiveListMember table contains lead information of active members belonging to a particular list.

Name	Type	Length	Is Required
id	int	18	Optional

listID	int	18	Optional
companyName	varchar	50	Optional
leadID	int	18	Optional
firstName	varchar	50	Optional
lastName	varchar	50	Optional
emailAddress	varchar	150	Optional

## Campaign

A campaign is a marketing project that we track in SharpSpring.

Name	Type	Length	Is Required
id	int	18	Optional
campaignName	varchar	255	Required
campaignType	varchar	255	Optional
campaignAlias	varchar	150	Optional
campaignOrigin	varchar	255	Optional
qty	int	18	Optional
price	double	18	Optional
goal	double	18	Optional
otherCosts	double	18	Optional
startDate	date		Optional
endDate	date		Optional
isActive	tinyint	1	Optional

## Client

The Client table contains information on Agency instances managed by the Partner instance.

Name	Type	Length	Is Required
id	int	18	Optional
companyName	varchar	255	Optional
streetAddress	varchar	255	Optional
zipCode	int	18	Optional
country	varchar	50	Optional
state	varchar	50	Optional
city	varchar	50	Optional



## CreateResult

CreateResult contains information about the success of a create operation, an array of any errors associated with the operation, and the ID of the newly created record.

Name	Type	Length	Is Required
success	tinyint	1	Optional
error	hash	255	Optional
id	int	18	Optional

## DealStage

A Deal Stage is a custom state for your Opportunity pipeline. As opportunities progress, the deal stage is changed until the Opportunity is finally won or lost.

Name	Type	Length	Is Required
id	int	18	Optional
dealStageName	varchar	255	Required
description	varchar	255	Optional
defaultProbability	double	18	Optional
weight	int	18	Optional
isEditable	tinyint	1	Optional

## DeleteResult

DeleteResult contains information about the success of a delete operation, and an array of any errors associated with the operation.

Name	Type	Length	Is Required
success	tinyint	1	Optional
error	hash	255	Optional

## Email

An html email created in SharpSpring.

Name	Type	Length	Is Required
id	int	18	Optional
html	text		Required
emailName	varchar	255	Required
subject	varchar	255	Required
fromName	varchar	255	Required

fromEmail	varchar	255	Required
replyTo	varchar	255	Optional
unsubCategory	int	18	Optional
contactManager	tinyint	1	Optional
repeatable	tinyint	1	Optional
fromLeadOwner	tinyint	1	Optional

## EmailEvent

Email Events contain information regarding particular lead's actions with particular emails sent in an email job.

Name	Type	Length	Is Required
leadID	int	18	Optional
emailID	int	18	Optional
emailJobID	int	18	Optional
createDate	timestamp	255	Optional
emailAddress	varchar	150	Optional

## EmailJob

An EmailJob associates data of a given jobID with emailIDs and recipientIDs.

Name	Type	Length	Is Required
id	int	18	Optional
emailID	int	18	Optional
isList	tinyint	1	Optional
isActive	tinyint	1	Optional
recipientID	int	18	Optional
sendCount	int	18	Optional
confirmedSend	tinyint	1	Optional
allowDuplicates	tinyint	1	Optional
createTimestamp	timestamp	255	Optional

## EmailListing

An Email Listing contains information about a group of emails, including the id, HTML content, subject line, and thumbnail image.

Name	Type	Length	Is Required
------	------	--------	-------------

id	int	18	Optional
createTimestamp	timestamp	255	Optional
title	varchar	255	Optional
subject	varchar	255	Optional
thumbnail	varchar	255	Optional

## EmailSend

EmailSend contains data about particular Email Jobs, such as opens, clicks, and unsubscribe metrics.

Name	Type	Length	Is Required
id	int	18	Optional
emailID	int	18	Optional
workflowID	int	18	Optional
sendCount	int	50	Optional
statistics	hash	255	Optional
domains	hash	255	Optional
numUnsubscribeReasons	array	50	Optional
uniqueOpens	int	50	Optional
uniqueClicks	int	50	Optional

## EmailSendEvent

EmailSendEvent contains information about actions a lead took with an email job.

Name	Type	Length	Is Required
id	int	18	Optional
contactID	int	18	Optional
type	varchar	50	Optional

## Event

The event table contains information related to SharpSpring user events, similar to the life-of-the-lead.

Name	Type	Length	Is Required
id	int	18	Optional
leadID	int	18	Required
eventName	varchar	255	Required

whatID	int	18	Optional
whatType	varchar	255	Optional
eventData	text	255	Optional
eventSource	text	255	Optional
createTimestamp	timestamp	255	Required

## Field

The field table contains metadata for a sharpspring field. This includes both lead and opportunity fields. The 'system name' is a key that can be specified in any POST to create or update an object.

Name	Type	Length	Is Required
id	int	18	Optional
relationship	varchar	255	Required
systemName	varchar	255	Optional
label	varchar	255	Required
source	varchar	255	Optional
dataType	varchar	255	Required
dataLength	int	18	Required
isRequired	tinyint	1	Required
isCustom	tinyint	1	Required
isActive	tinyint	1	Required
isAvailableInContactManager	tinyint	1	Required
isEditableInContactManager	tinyint	1	Required
isAvailableInForms	tinyint	1	Required

## FormEventHistory

FormEventHistory ties emailAddresses to trackingIDs. It is useful for de-duping contacts in the CRM.

Name	Type	Length	Is Required
id	int	18	Optional
trackingID	varchar	50	Optional
emailAddress	varchar	255	Optional
createTimestamp	timestamp	255	Optional

## Lead

The Lead table consists of prospects who are possibly interested in your product. As a lead progresses through your pipeline, their status changes from unqualified to qualified. A lead can be converted into a contact, opportunity, or account.

If you wish to update custom fields for a lead, specify the custom field's "systemName" as the key in your 'lead' object. In order to get a list of system names, first retrieve members of the Field table.

Name	Type	Length	Is Required
id	bigint	20	Optional
accountID	bigint	20	Optional
ownerID	bigint	20	Optional
campaignID	bigint	20	Optional
leadStatus	picklist		Optional
leadScore	smallint	6	Optional
leadScoreWeighted	smallint	6	Optional
persona	string	128	Optional
active	tinyint	1	Optional
firstName	varchar	50	Optional
lastName	varchar	50	Optional
emailAddress	varchar	150	Required
companyName	varchar	100	Optional
title	varchar	255	Optional
street	varchar	255	Optional
city	varchar	255	Optional
country	varchar	255	Optional
state	varchar	50	Optional
zipcode	varchar	50	Optional
website	varchar	255	Optional
phoneNumber	varchar	255	Optional
trackingID	varchar	255	Optional
officePhoneNumber	varchar	255	Optional
phoneNumberExtension	varchar	255	Optional
mobilePhoneNumber	varchar	255	Optional
faxNumber	varchar	255	Optional
description	text		Optional
industry	varchar	255	Optional
isUnsubscribed	tinyint	1	Optional
updateTimestamp	timestamp		Optional

## List

List is a segmented audience of leads in SharpSpring.

Name	Type	Length	Is Required
id	int	18	Optional
name	varchar	255	Optional
memberCount	int	18	Optional
removedCount	int	18	Optional
createTimestamp	timestamp	255	Optional
description	varchar	255	Optional

## ListMember

ListMember is a mapping table that corresponds leads in SharpSpring to the list they are members of.

Name	Type	Length	Is Required
id	int	18	Optional
listID	int	18	Optional
memberID	int	18	Optional
isRemoved	tinyint	1	Optional

## ListMembership

ListMembership shows all of the Lists, by id, a lead is a member of as well as when they were added to the list.

Name	Type	Length	Is Required
id	int	18	Optional
name	varchar	255	Optional
status	varchar	50	Optional
createTimestamp	timestamp	255	Optional

## ListTag

ListTag is a table of all list tags in an instance of SharpSpring. List tags are used to group similar lists together for aggregate list sends, automation, and reporting.

Name	Type	Length	Is Required
id	int	18	Optional
label	varchar	50	Optional
objectType	varchar	50	Optional

## Opportunity

An Opportunity represents a potential deal that has an expected value. An opportunity can be associated with a Lead or Account, as well as a Campaign.

Name	Type	Length	Is Required
id	int	18	Optional
ownerID	int	18	Required
dealStageID	int	18	Required
accountID	int	18	Optional
campaignID	int	18	Optional
opportunityName	varchar	255	Required
probability	double	18	Optional
amount	double	18	Optional
isClosed	tinyint	1	Optional
isWon	tinyint	1	Optional
isActive	tinyint	1	Optional
closeDate	timestamp	255	Required
originatingLeadID	int	18	Optional
primaryLeadID	int	18	Optional

## OpportunityLead

OpportunityLead is a mapping table that represents an association between an Opportunity and Lead. Each Opportunity in SharpSpring can consist of multiple Leads.

Name	Type	Length	Is Required
id	int	18	Optional
opportunityID	int	18	Required
leadID	int	18	Required

## RemovedListMember

RemovedListMember is a table containing information on leads removed from a particular list.

Name	Type	Length	Is Required
listMemberID	int	18	Optional
listID	int	18	Optional
isRemoved	tinyint	1	Optional
isUnsubscribed	tinyint	1	Optional

hardBounced	tinyint	1	Optional
leadID	int	18	Optional
firstName	varchar	50	Optional
lastName	varchar	50	Optional
emailAddress	varchar	150	Optional

## ScheduleSend

An email send that is scheduled to be performed in SharpSpring A listID or listTagID must be specified. The send must either be scheduled to send immediately or a date/time must be provided.

Name	Type	Length	Is Required
emailID	int	18	Required
listID	int	18	Required
listTagID	int	18	Required
immediate	tinyint	1	Optional
date	date	255	Optional
hours	int	18	Optional
minutes	int	18	Optional
meridiem	varchar	255	Optional
sendDuplicate	tinyint	1	Optional
suppressAtRisk	tinyint	1	Optional
suppressUnengaged	tinyint	1	Optional

## ScheduleSendQueue

ScheduleSendQueue contains information about email jobs sending in the Schedule Events queue.

Name	Type	Length	Is Required
id	int	18	Optional
active	tinyint	1	Optional
fromemail	varchar	150	Optional
fromname	varchar	50	Optional
senddate	timestamp	255	Optional
startdate	timestamp	255	Optional
subject	varchar	150	Optional



## UnsubscribeCategory

The UnsubscribeCategory table consists of the different email categorizations that SharpSpring users can label emails with and categories that recipients can choose to unsubscribe from.

Name	Type	Length	Is Required
id	int	18	Optional
companyProfileID	int	18	Optional
label	varchar	50	Optional

## UpdateResult

UpdateResult contains information about the success of an update operation, and an array of any errors associated with the operation.

Name	Type	Length	Is Required
success	tinyint	1	Optional
error	hash	255	Optional

## UserProfile

The User Profile table consists of SharpSpring user accounts. This ID can be used in the lead owner field in the Lead table as well as in the Opportunity table.

Name	Type	Length	Is Required
id	int	18	Optional
firstName	varchar	50	Required
lastName	varchar	50	Required
displayName	varchar	100	Required
emailAddress	varchar	100	Required
isActive	tinyint	1	Required
isReseller	tinyint	1	Optional
userTimezone	varchar	255	Optional
phone	varchar	255	Optional

# SharpSpring APIv1.117

Major Versions: [v1](#)

Overview	Example Code	Methods	Schema	Error Codes
----------	--------------	---------	--------	-------------

## Error Codes

### Codes

Error objects can be returned in the case where an API level error such as an invalid login is supplied or where an object-level error such as an invalid field is supplied.

In the case of object-level errors, the request has succeeded so no request-level errors will be supplied

Message	Code
Invalid request data format	101
Header missing request ID	102
Header missing account ID	103
Header missing Secret Key	104
User authentication failed	105
Exceeded daily method call limit	106
Missing method signature	201
Missing params array	202
Missing header array	203
Invalid method signature	204
Invalid parameters	205
Missing parameters	206
Missing objects array	207
Object is malformed (expected array, got scalar)	208
Attempting to execute an excess amount of queries	209
Got array, expected variable	210
Only custom fields can be deleted	211
Object is not owned by the company	212
The lead you tried to add is already a member	213
The lead you tried to add does not exist for this account	214
You have supplied an invalid country code.	215
Company not found. Please make sure the ID supplied belongs to your company	216
Incorrect parameter value supplied.	217
Invalid username supplied, needs to be in the form of a email address	218

Incorrect eventSource value supplied.	219
Please supply a valid whoID	220
The current username / email address is already in use by another account.	221
You have exceeded the amount of leads to process. A max of 100 leads is allowed	222
Please make sure to add all relevant fields for a user is supplied: userEmail, userPassword, userDisplayName	223
Entry already exists	301
No table rows affected	302
Cannot add or update a child row: a foreign key constraint fails	303
No sub-accounts supplied	304
Permission denied, you are not a reseller.	305
Permission denied, you do not manage these accounts.	306
Sub account not found	307
A record with that ID does not exist	404
Unknown error. Please contact Sharpspring Developer Support.	999