

Table of Contents

API Reference.....	15
What is API?	15
Benefits	15
Examples.....	15
Campaign.Create.....	17
Description.....	17
Request Parameters.....	17
Response Parameters.....	17
Error Codes	17
Campaign.Update.....	18
Description.....	18
Request Parameters.....	18
Error Codes	19
Campaigns.Get	20
Description.....	20
Request Parameters.....	20
Response Parameters.....	21
Error Codes	22
Campaign.Get	22
Description.....	22
Request Parameters.....	22
Response Parameters.....	23
Error Codes	23
Campaigns.Delete.....	23
Description.....	23
Request Parameters.....	23
Response Parameters.....	24
Error Codes	24
Campaigns.Archive.GetURL	24

- Description**..... 24
- Request Parameters** 24
- TemplateURL = {string} URL of the public archive page template (optional) 25
- Response Parameters**..... 25
- Error Codes** 25
- Client.Login** 25
 - Description**..... 25
 - Request Parameters**..... 25
 - Response Parameters**..... 26
 - Error Codes** 26
- Client.PasswordRemind**..... 27
 - Description**..... 27
 - Request Parameters**..... 27
 - Response Parameters**..... 27
 - Error Codes** 27
- Client.PasswordReset**..... 28
 - Description**..... 28
 - Request Parameters**..... 28
 - Response Parameters**..... 28
 - Error Codes** 28
- Client.Create**..... 29
 - Description**..... 29
 - Request Parameters**..... 29
 - Response Parameters**..... 29
 - Error Codes** 29
- Client.Update**..... 30
 - Description**..... 30
 - Request Parameters**..... 30
 - Response Parameters**..... 31
 - Error Codes** 31
- Clients.Get** 32
 - Description**..... 32

- Request Parameters** 32
- Response Parameters** 32
- Error Codes** 33
- Clients.Delete** 33
 - Description** 33
 - Request Parameters** 33
 - Response Parameters** 33
 - Error Codes** 34
- Client.AssignCampaigns** 34
 - Description** 34
 - Request Parameters** 34
 - Response Parameters** 34
 - Error Codes** 35
- Client.AssignSubscriberLists** 35
 - Description** 35
 - Request Parameters** 35
 - Response Parameters** 36
 - Error Codes** 36
- Client.Lists.Get** 36
 - Description** 36
 - Request Parameters** 36
 - Response Parameters** 36
 - Error Codes** 37
- Client.List.Get** 37
 - Description** 37
 - Request Parameters** 37
 - Response Parameters** 38
 - Error Codes** 38
- CustomField.Create** 38
 - Description** 38
 - Request Parameters** 38
 - Response Parameters** 40

- Error Codes** 40
- CustomField.Update** 40
 - Description**..... 40
 - Request Parameters** 40
 - Response Parameters**..... 42
 - Error Codes** 42
- CustomFields.Copy** 43
 - Description**..... 43
 - Request Parameters** 43
 - Response Parameters**..... 43
 - Error Codes** 43
- CustomFields.Get** 44
 - Description**..... 44
 - Request Parameters** 44
 - Response Parameters**..... 44
 - Error Codes** 45
- CustomFields.Delete** 45
 - Description**..... 45
 - Request Parameters** 45
 - Response Parameters**..... 46
 - Error Codes** 46
- Email.Create** 46
 - Description**..... 46
 - Request Parameters** 46
 - Response Parameters**..... 46
- Email.Delete**..... 47
 - Description**..... 47
 - Request Parameters** 47
 - Response Parameters**..... 47
 - Error Codes** 47
- Email.Get** 48
 - Description**..... 48

- Request Parameters** 48
- Response Parameters** 48
- Error Codes** 48
- Emails.Get** 49
 - Description** 49
 - Request Parameters** 49
 - Response Parameters** 49
 - Error Codes** 49
- Email.Update** 50
 - Description** 50
 - Request Parameters** 50
 - Response Parameters** 51
 - Error Codes** 51
- Email.Duplicate** 52
 - Description** 52
 - Request Parameters** 52
 - Response Parameters** 53
 - Error Codes** 53
- Email.PersonalizationTags** 53
 - Description** 53
 - Request Parameters** 53
 - Response Parameters** 54
 - Error Codes** 54
- Email.EmailPreview** 54
 - Description** 54
 - Request Parameters** 55
 - Response Parameters** 55
 - Error Codes** 55
- Email.SpamTest** 56
 - Description** 56
 - Request Parameters** 56
 - Response Parameters** 56

Error Codes 57

Email.Template.Create 57

Description..... 57

Request Parameters 57

Response Parameters..... 58

Error Codes 58

Email.Template.Get 59

Description..... 59

Request Parameters 59

Response Parameters..... 59

Error Codes 59

Email.Templates.Get 60

Description..... 60

Request Parameters 60

Response Parameters..... 60

Error Codes 60

Email.Template.Update..... 61

Description..... 61

Request Parameters 61

Response Parameters..... 62

Error Codes 62

Email.Template.Delete 62

Description..... 62

Request Parameters 62

Response Parameters..... 63

Error Codes 63

ListIntegration.AddURL..... 63

Description..... 63

Request Parameters 63

Response Parameters..... 64

Error Codes 64

ListIntegration.GetURLs..... 65

- Description..... 65**
- Request Parameters 65**
- Response Parameters..... 65**
- Error Codes 65**
- ListIntegration.TestURL 66**
 - Description..... 66**
 - Request Parameters 66**
 - Response Parameters..... 66**
 - Error Codes 66**
- ListIntegration.DeleteURLs 67**
 - Description..... 67**
 - Request Parameters 67**
 - Response Parameters..... 67**
 - Error Codes 67**
- ListIntegration.GenerateSubscriptionFormHTMLCode 68**
 - Description..... 68**
 - Request Parameters 68**
 - Response Parameters..... 68**
 - Error Codes 69**
- ListIntegration.GenerateUnsubscriptionFormHTMLCode 69**
 - Description..... 69**
 - Request Parameters 69**
 - Response Parameters..... 69**
 - Error Codes 70**
- List.Create..... 70**
 - Description..... 70**
 - Request Parameters 70**
 - Response Parameters..... 71**
 - Error Codes 71**
- List.Update..... 71**
 - Description..... 71**
 - Request Parameters 71**

- Response Parameters**..... 72
- Error Codes** 73
- Lists.Get** 73
 - Description**..... 73
 - Request Parameters**..... 73
 - Response Parameters**..... 74
 - Error Codes** 74
- List.Get**..... 74
 - Description**..... 74
 - Request Parameters**..... 75
 - Response Parameters**..... 75
 - Error Codes** 75
- Lists.Delete** 76
 - Description**..... 76
 - Request Parameters**..... 76
 - Response Parameters**..... 76
 - Error Codes** 76
- Media.Upload**..... 77
 - Description**..... 77
 - Request Parameters**..... 77
 - Response Parameters**..... 77
 - Error Codes** 78
- Media.Browse**..... 78
 - Description**..... 78
 - Request Parameters**..... 78
 - Response Parameters**..... 78
 - Error Codes** 79
- Media.Retrieve** 79
 - Description**..... 79
 - Request Parameters**..... 79
 - Response Parameters**..... 80
 - Error Codes** 80

Media.Delete 80

Description..... 80

Request Parameters..... 80

Response Parameters..... 81

Error Codes 81

Segment.Create 81

Description..... 81

Request Parameters..... 81

Response Parameters..... 82

Error Codes 82

Segment.Update 82

Description..... 82

Request Parameters..... 82

Response Parameters..... 83

Error Codes 83

Segment.Get 84

Description..... 84

Request Parameters..... 84

Response Parameters..... 84

Error Codes 85

Segments.Delete 85

Description..... 85

Request Parameters..... 85

Response Parameters..... 85

Error Codes 86

Segments.Copy 86

Description..... 86

Request Parameters..... 86

Response Parameters..... 86

Error Codes 87

Subscriber.Login..... 87

Description..... 87

Request Parameters 87

Response Parameters 88

Error Codes 88

Subscriber.GetLists 89

Description 89

Request Parameters 89

Response Parameters 89

Error Codes 89

Subscriber.Get 90

Description 90

Request Parameters 90

Response Parameters 90

Error Codes 91

Subscribers.Get 91

Description 91

Request Parameters 91

Response Parameters 92

Error Codes 92

Subscriber.Subscribe 93

Description 93

Request Parameters 93

Response Parameters 93

Error Codes 94

Subscriber.OptIn 95

Description 95

Request Parameters 95

Response Parameters 95

Error Codes 96

Subscriber.Unsubscribe 96

Description 96

Request Parameters 96

Response Parameters 97

- Error Codes** 97
- Subscriber.Update** 98
 - Description**..... 98
 - Request Parameters** 98
 - Response Parameters**..... 99
 - Error Codes** 99
- Subscribers.Delete** 100
 - Description**..... 100
 - Request Parameters** 100
 - Response Parameters**..... 101
 - Error Codes** 101
- Settings.Update** 101
 - Description**..... 101
 - Request Parameters** 101
 - Default_Optin_Email_Subject = {string} Default_Optin_Email_Body = {string} 106
 - Response Parameters**..... 106
 - Error Codes** 106
- Settings.EmailSendingTest** 107
 - Description**..... 107
 - Request Parameters** 107
 - Response Parameters**..... 107
 - Error Codes** 108
- Tag.Create** 108
 - Description**..... 108
 - Request Parameters** 108
 - Response Parameters**..... 108
 - Error Codes** 109
- Tags.Get** 109
 - Description**..... 109
 - Request Parameters** 109
 - Response Parameters**..... 109
 - Error Codes** 110

Tag.Update..... 110

Description..... 110

Request Parameters..... 110

Response Parameters..... 111

Error Codes 111

Tags.Delete 111

Description..... 111

Request Parameters..... 111

Response Parameters..... 111

Error Codes 112

Tag.AssignToCampaigns..... 112

Description..... 112

Request Parameters..... 112

Response Parameters..... 113

Error Codes 113

Tag.UnassignFromCampaigns..... 113

Description..... 113

Request Parameters..... 113

Response Parameters..... 114

Error Codes 114

User.Get..... 114

Description..... 114

Request Parameters..... 114

Response Parameters..... 115

Error Codes 115

User.Login..... 115

Description..... 115

Request Parameters..... 115

Response Parameters..... 116

Error Codes 116

User.PasswordRemind..... 117

Description..... 117

Request Parameters 117

Response Parameters 117

Error Codes 117

User.PasswordReset 118

Description 118

Request Parameters 118

Response Parameters 118

Error Codes 118

Theme.Create 119

Description 119

Request Parameters 119

Response Parameters 119

Error Codes 120

Theme.Get 120

Description 120

Request Parameters 120

Response Parameters 120

Error Codes 121

Theme.Update 121

Description 121

Request Parameters 121

Response Parameters 121

Error Codes 122

Theme.Delete 122

Description 122

Request Parameters 122

Response Parameters 123

Error Codes 123

Themes.Get 123

Description 123

Request Parameters 123

Response Parameters 124

Error Codes 124

API Reference

What is API?

[API](#) allows developers to integrate different softwares easily. Imnica Mail includes full-featured API. You can access to API function of almost every functionality that you can see on the Imnica Mail user interface.

Imnica Mail's API allows you to connect and integrate via HTTP POST and GET methods.

To give you an example, you can integrate your website shopping cart with Imnica Mail easily. For every new customer, you can call Imnica Mail's subscription API command and subscribe the new customer email address to a subscriber list and let Imnica Mail to send your customer a series of post-purchase auto responders.

Imnica Mail API gives you endless capabilities and opportunities.

Benefits

We tried to list a few benefits that you will have with Imnica Mail's API capabilities:

- Full-featured API commands
- Integrate with any third party desktop or web based application
- Expand Imnica Mail's capabilities and feature sets
- Simple API communication
- Ability to receive response data in [JSON](#) or [XML](#) format

Examples

Sending Request To Imnica Mail API

Imnica Mail API can accept data in HTTP POST or HTTP GET method. Below, we provided an HTTP GET example that logs you into user area:

```
http://www.imnicamail.com/v4/api.php?Command=User.Login&ResponseFormat=XML&Username=admin&Password=admin
```

API Response

Once you execute the above API call, you will receive either a success data or error data:

On error:

```
<?xml version="1.0" encoding="utf-8"?>
<response>
  <Success></Success>
  <ErrorCode/>
  <ErrorText/>
</response>
```

On success:

```
<?xml version="1.0" encoding="utf-8"?>
<response>
  <Success><![CDATA[1]]></Success>
  <ErrorCode><![CDATA[0]]></ErrorCode>
  <ErrorText></ErrorText>
  <SessionID><![CDATA[08be000c325a0cc466d85032bf9c9573]]></SessionID>
  <UserInfo>
    <UserID><![CDATA[1]]></UserID>
    <RelUserGroupID><![CDATA[1]]></RelUserGroupID>
    <EmailAddress><![CDATA[test@domain.com]]></EmailAddress>
    <Username><![CDATA[admin]]></Username>
    <Password><![CDATA[21232f297a57a5a743894a0e4a801fc3]]></Password>
  </UserInfo>
</response>
```


Campaign.Create

Description

Create new campaign to send out

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Campaign.Create} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

CampaignName = {string} (required)

Name of the campaign

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

CampaignID = {integer}

ID of the new campaign record

Error Codes

1 - Campaign name is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Campaign.Update

Description

Update campaign details

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Campaign.Update} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

CampaignID = {integer} (required)

Campaign ID to update

CampaignStatus = {Draft | Ready | Sending | Paused | Pending Approval | Sent | Failed} (required)

Set the campaign status

CampaignName = {string} (required)

Name of the campaign

RelEmailID = {integer} (required)

The email content of campaign (email ID)

ScheduleType = {Not Scheduled | Immediate | Future | Recursive} (required)

Type of the schedule

SendDate = {YYYY-MM-DD} (required)

Date to send campaign

SendTime = {HH: MM: SS} (required)
Time to send campaign

SendTimeZone = {string} (required)
Time zone of the schedule date

ScheduleRecDaysOfWeek = {0 | 1 | 2 | 3 | 4 | 5 | 6 | 7} (required)
(Recursive scheduling) separate values with comma (.). Enter 0 for every day

ScheduleRecDaysOfMonth = {0 | 1 | 2 | 3 | 4 | 5 | ... | 31} (required)
(Recursive scheduling) separate with comma (.). Enter 0 for every day

ScheduleRecMonths = {0 | 1 | 2 | ... | 12} (required)
(Recursive scheduling) separate with comma (.). Enter 0 for every month

ScheduleRecHours = {0 | 1 | 2 | ... | 23} (required)
(Recursive scheduling) separate with comma (.)

ScheduleRecMinutes = {0 | 1 | 2 | ... | 59} (required)
(Recursive scheduling) separate with comma (.)

ScheduleRecSendMaxInstance = {integer} (required)
(Recursive scheduling) number of times to repeat campaign sending (enter 0 for unlimited)

ApprovalUserExplanation = {string} (required)
User explanation for the campaign if campaign is pending for approval

GoogleAnalyticsDomains = {string} (required)
Domains to track with Google Analytics (seperate domains with line break (\n))

RecipientListsAndSegments = {string} (required)
Target subscriber lists and segments. Each segment and list is seperated by comma. Format: ListID: SegmentID,ListID: SegmentID Ex: 3: 0,3: 2

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - Missing campaign ID
- 2 - Invalid campaign ID
- 3 - Invalid campaign status value
- 4 - Invalid email ID
- 5 - Invalid campaign schedule type value
- 6 - Missing send date
- 7 - Missing send time
- 8 - Day of month or day of week must be provided for recursive scheduling
- 9 - Months must be provided for recursive scheduling
- 10 - Hours must be provided for recursive scheduling
- 11 - Minutes must be provided for recursive scheduling
- 12 - Number of times to repeat must be provided for recursive scheduling
- 13 - Time zone for scheduling is missing
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Campaigns.Get

Description

Retrieves campaigns of a user.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Campaigns.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

OrderField = {string} (required)
Name of the field to order based on

OrderType = {ASC | DESC} (required)
Ascending or descending ordering

CampaignStatus = {All | Draft | Ready | Sending | Paused | Pending Approval | Sent | Failed} (required)
Status of campaigns to retrieve

SearchKeyowrd = {string} (required)
Keyword to look for in CampaignName field

RecordsPerRequest = {integer} (required)
How many rows to return per page

RecordsFrom = {integer} (required)
Start from (starts from zero)

RetrieveTags = {true|false} (v4.0.4+)
Set to true if you are going to filter campaign list based on specific filters

Tags = {string} (v4.0.4+)
Tag ID numbers separated with comma if you are going to filter campaign list based on tags

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Campaigns = {array}
Returns the list of all campaigns in array

TotalCampaigns = {integer}
Returns the total number of campaigns

Error Codes

1 - Missing user ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Campaign.Get

Description

Retrieve a specific campaign of a user.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Campaign.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

CampaignID = {integer} (required)
Campaign id

RetrieveStatistics = {true | false} (required)
If set to true, returns campaign statistics too

RetrieveTags = {true|false} (v4.0.4+)

Set to true if you want to return the list of assigned tags also

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Campaign = {array}
Campaign information

Error Codes

1 - Missing campaign ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Campaigns.Delete

Description

Deletes given campaigns.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Campaigns.Delete} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Campaigns = {string} (required)
Comma delimited Campaign ids

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Campaign ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Campaigns.Archive.GetURL

Description

Returns the URL for the public archive page of campaigns

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Campaigns.Archive.GetURL} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

TagID = {integer} (required)

Target tag ID to retrieve campaigns

TemplateURL = {string}

URL of the public archive page template (optional)

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

URL = {string}

URL for the public archive page

Error Codes

1 - Tag ID is missing

2 - Tag does not exist

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.Login

Description

Verifies and logs in the client account

Request Parameters

Command = {Client.Login} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Username = {string} (required)
Username of the client to be logged in

Password = {string} (required)
Password of the client to be logged in

RememberMe = {string} (required)
Stores the login information in the cookie

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

SessionID = {string}
SessionID of the logged in user

Error Codes

1 - Username is missing

2 - Password is missing

3 - Invalid login information

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.PasswordRemind

Description

Sends the password reset email.

Request Parameters

Command = {Client.PasswordRemind} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailAddress = {string} (required)
Email address of the client to be reminded

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - Email address is missing
- 2 - Invalid email address
- 3 - Email address not found in client database
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Client.PasswordReset

Description

Resets clients password and sends the new password with email.

Request Parameters

Command = {Client.PasswordReset} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ClientID = {integer} (required)

ID of the client whose password will be reset

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - Client id is missing

2 - Invalid user id

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.Create

Description

Create new client account

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Client.Create} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

ErrorMessage = {text}

Error Codes

- 1 - Client name missing
- 2 - Client username missing
- 3 - Client password missing
- 4 - Client email address missing
- 5 - Email address format is invalid

6 - Username already registered to another client

7 - Email address already registered to another client

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.Update

Description

Update client account information

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Client.Update} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Access = {user | client} (required)

Determines the user type of api command. Default is user.

ClientID = {integer} (required)

ID number of the client account

ClientUsername = {string} (required)

Username of the client account

ClientPassword = {string} (required)

Password of the client account

ClientEmailAddress = {string} (required)
Email address of the client account

ClientName = {string} (required)
Name of the client account

ClientAccountStatus = {Enabled | Disabled} (required)
Account status of the client

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Error Codes

- 1 - Client name is missing
- 2 - Client username is missing
- 3 - Client password is missing
- 4 - Client email address is missing
- 5 - Invalid client account status
- 6 - Client ID is missing
- 7 - Email address format is invalid
- 8 - Insufficient privileges for updating the client account
- 9 - Username already registered to another client
- 10 - Email address already registered to another client
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Clients.Get

Description

Retrieves clients of logged in user.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Clients.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

OrderField = {any client field} (required)

Order field

OrderType = {ASC | DESC} (required)

Order type

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

TotalClientCount = {mixed}

Total number of clients user has

Clients = {mixed}

Returned clients

Error Codes

1 - order by field missing

2 - sort type is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Clients.Delete

Description

Deletes given clients.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Clients.Delete} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Clients = {string} (required)

Comma delimited Client ids

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Client ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.AssignCampaigns

Description

Assigns given campaigns to client.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Client.AssignCampaigns} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ClientID = {integer} (required)

ID of client

CampaignIDs = {string} (required)

Comma seperated campaign ids to be assigned

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Client id is missing

2 - Campaign ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.AssignSubscriberLists

Description

Assigns given subscriber lists to client.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Client.AssignSubscriberLists} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ClientID = {integer} (required)
ID of client

SubscriberListIDs = {string} (required)
Comma separated subscriber list ids to be assigned

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Client id is missing

2 - Subscriber list ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.Lists.Get

Description

Retrieves assigned subscriber lists of logged in client.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Client.Lists.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

TotalListCount = {integer}
Total number of subscriber lists

Lists = {mixed}
Returned lists

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Client.List.Get

Description

Retrieves subscriber list of logged in client.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Client.List.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer} (required)
ID of subscriber list to retrieve

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

List = {mixed}
Returned list information

Error Codes

1 - Subscriber list id is missing

2 - Client account does not have permission to view this list's information

99998 - Authentication failure or session expired

99999 - Not enough privileges

CustomField.Create

Description

Creates a new custom field for given subscriber list.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {CustomField.Create} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer} (required)
ID of subscriber list

PresetName = {Gender | Age | Employment | Income | Education | Days of the week | Months of the year | U.S. states | Continents | Satisfaction | Importance | Agreement | Comparison}

You can create a new custom field by selecting a preset. This way all options of the preset will be created by backend.

FieldName = {string} (required)
Name of new custom field

FieldType = {Single line | Paragraph text | Multiple choice | Drop down | Checkboxes | Hidden field} (required)
Type of new custom field

FieldDefaultValue = {string} (required)
Default value of new custom field

OptionLabel[n] = {string} (required)
label of nth option

OptionValue[n] = {string} (required)
value of nth option

OptionSelected = {array} (required)
array of selected option ids (Option ids are n)

ValidationMethod = {Disabled | Numbers | Letters | Numbers and letters | Email address | URL | Date | Time | Custom} (required)
Validation method of custom field

ValidationRule = {string} (required)
Validation rule of custom field

Visibility = {Public | User Only}
Whether to show custom field in subscriber area or not

IsRequired = {Yes | No}
Whether to make the custom field mandatory to fill in or not

IsUnique = {Yes | No}
Whether to force custom field entry to be unique or not

IsGlobal = {Yes | No}

If this parameter is set to 'Yes', custom field will be valid for all lists in the user account

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

CustomFieldID = {integer}

ID of new custom field

Error Codes

1 - Subscriber list id is missing

2 - Field name is missing

3 - Field type is missing

4 - Validation rule is missing

5 - Invalid custom field preset name

99998 - Authentication failure or session expired

99999 - Not enough privileges

CustomField.Update

Description

Updates custom field information.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {CustomField.Update} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

CustomFieldID = {integer} (required)

ID of custom field to be updated

FieldName = {string} (required)

Name of new custom field

FieldType = {Single line | Paragraph text | Multiple choice | Drop down | Checkboxes | Hidden field} (required)

Type of new custom field

FieldDefaultValue = {string} (required)

Default value of new custom field

OptionLabel[n] = {string} (required)

label of nth option

OptionValue[n] = {string} (required)

value of nth option

OptionSelected = {array} (required)

array of selected option ids (Option ids are n)

ValidationMethod = {Disabled | Numbers | Letters | Numbers and letters | Email address | URL | Date | Time | Custom} (required)

Validation method of custom field

ValidationRule = {string} (required)

Validation rule of custom field

Visibility = {Public | User Only}

Whether to show custom field in subscriber area or not

IsRequired = {Yes | No}

Whether to make the custom field mandatory to fill in or not

IsUnique = {Yes | No}

Whether to force custom field entry to be unique or not

IsGlobal = {Yes | No}

If this parameter is set to 'Yes', custom field will be valid for all lists in the user account

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - Custom field id is missing

2 - Field name is missing

3 - Field type is missing

4 - Validation rule is missing

6 - Custom field id is invalid

7 - Field type is invalid

8 - Validation method is invalid

9 - Invalid visibility method

10 - Invalid IsRequired value

11 - Invalid IsUnique value

12 - Invalid IsGlobal value

99998 - Authentication failure or session expired

99999 - Not enough privileges

CustomFields.Copy

Description

Copies custom fields of a subscriber list to another subscriber list.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {CustomFields.Copy} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SourceListID = {integer} (required)

ID of source subscriber list

TargetListID = {integer} (required)

ID of target subscriber list

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

- 1 - Source subscriber list id is missing
- 2 - Target subscriber list id is missing
- 3 - Source subscriber list id is invalid

4 - Target subscriber list id is invalid

99998 - Authentication failure or session expired

99999 - Not enough privileges

CustomFields.Get

Description

Retrieves custom fields of given subscriber list.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {CustomFields.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

OrderField = {field name of custom field} (required)

Order field

OrderType = {ASC | DESC} (required)

Order type

SubscriberListID = {integer} (required)

Subscriber list id

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code
TotalCustomFields = {integer}
Total number of custom fields of subscriber list

CustomFields = {mixed}
Returned custom fields

Error Codes

1 - Subscriber list id is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

CustomFields.Delete

Description

Deletes given custom fields.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {CustomFields.Delete} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

CustomFields = {string} (required)
Comma delimited custom field ids.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Custom field ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Create

Description

Creates a blank email record for user.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Create} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Email.Delete

Description

Deletes given email.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Delete} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailID = {integer} (required)
Id of email to be deleted.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Email id is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Get

Description

Retrieves all information of given email.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailID = {integer} (required)

Email id

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - Email id is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Emails.Get

Description

Returns the list of email contents created so far

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Emails.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

TotalEmailCount = {integer}

Number of email contents returned

Emails = {array}

List of emails in array format

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Update

Description

Updates custom field information.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Update} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ValidateScope = {OptIn | Campaign | AutoResponder} (required)

Defines the validation scope of the email (required fields, links, etc.)

EmailID = {integer} (required)

Id of the email to be updated

EmailName = {string} (required)

Name of the email

FromName = {string} (required)

From name of the email

FromEmail = {string} (required)

From email address of the email

ReplyToName = {string} (required)

Reply to name of the email

ReplyToEmail = {string} (required)
Reply to email address of the email

Mode = {Empty | Template | Import} (required)
Email's content mode

FetchURL = {string} (required)
Email's remote content url

Subject = {string} (required)
Email's subject

PlainContent = {string} (required)
Email's plain content

HTMLContent = {string} (required)
Email's html content

ImageEmbedding = {Enabled | Disabled} (required)
Email's image embedding

RelTemplateID = {integer} (required)
Email's template id

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - Email id is missing
- 2 - FetchURL is missing
- 3 - Email id is invalid
- 4 - Mode is invalid
- 5 - RelTemplateID is missing
- 6 - FromEmail email address is invalid

- 7 - ReplyToEmail email address is invalid
- 8 - Plain and HTML content is empty
- 9 - Missing validation scope parameter
- 10 - Invalid validation scope parameter
- 11 - Missing unsubscription link in HTML content
- 12 - Missing unsubscription link in plain content
- 13 - Missing opt-in confirmation link in HTML content
- 14 - Missing opt-in confirmation link in plain content
- 15 - Missing opt-in reject link in HTML content
- 16 - Missing opt-in reject link in plain content
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Email.Duplicate

Description

Creates a new email content based on existing one

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Duplicate} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

EmailID = {integer} (required)

The ID number of email content which is going to be duplicated

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

EmailID = {integer}

ID number of the new email

EmailName = {string}

Name of the new email

Error Codes

1 - Email ID is missing

2 - Email not found

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.PersonalizationTags

Description

Returns the list of personalization tags

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.PersonalizationTags} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer | array} (required)
If the Scope is 'Subscriber', list ID must be provided. It can be an integer single list ID or multiple list IDs in array
Scope = {array} (required)
Types of tags to return. Possible array values: 'Subscriber', 'CampaignLinks', 'OptLinks', 'ListLinks', 'AllLinks', 'User'

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - Scope is missing
- 2 - List ID is missing
- 3 - List ID does not belong to the logged in user
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Email.EmailPreview

Description

Sends a preview email to the provided email address

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.EmailPreview} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailID = {integer} (required)

Email ID number

EmailAddress = {string} (required)

Email address to send preview of the email

ListID = {integer} (required)

List ID number

CampaignID = {integer} (required)

Campaign ID number

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - Missing email ID

2 - Missing email address

3 - Invalid email ID

4 - Invalid email address format

5 - Missing list ID

6 - Invalid list ID

7 - Invalid campaign ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.SpamTest

Description

Returns SpamAssassin spam filter test results of your email

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.SpamTest} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailID = {integer} (required)

Email ID number

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

TestResults = {array}

Test results will be returned in JSON array format

Error Codes

1 - Missing email ID

2 - Invalid email ID

3 - License error

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Template.Create

Description

Creates a new email template

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Template.Create} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

RelOwnerUserID = {integer} (required)

Assigned user (set to 0 to make visible to anyone)

TemplateName = {string} (required)

Name of the email template

TemplateDescription = {string} (required)
Description of the email template

TemplateSubject = {string} (required)
Subject of the email template

TemplateHTMLContent = {string} (required)
HTML content of the email template

TemplatePlainContent = {string} (required)
Plain content of the email template

TemplateThumbnailPath = {string} (required)
Thumbnail path of the template on the server

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

TemplateID = {integer}
ID of the new email template

Error Codes

1 - Template name is missing

2 - At least one of the email content types must be provided (HTML, plain or both)

3 - Target user is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Template.Get

Description

Retrieves email template requested by the user

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Template.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

TemplateID = {integer} (required)

ID of the requested template

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Template = {array}

list of available templates

Error Codes

1 - Template ID is missing

2 - Template ID is invalid

3 - Template does not belong to this user

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Templates.Get

Description

Retrieves email templates defined in the system

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Templates.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

UserID = {integer} (required)

Templates of a specific user (administrator only)

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Templates = {array}

list of available templates

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Template.Update

Description

Updates email template

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Template.Update} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

TemplateID = {string} (required)

Template ID to update

TemplateName = {string} (required)

Name of the email template

TemplateDescription = {string} (required)

Description of the email template

TemplateSubject = {string} (required)

Subject of the email template

TemplateHTMLContent = {string} (required)

HTML content of the email template

TemplatePlainContent = {string} (required)
Plain content of the email template
TemplateThumbnailPath = {string} (required)
Thumbnail path of the template on the server

RelOwnerUserID = {string} (required)
Owner user ID of the template

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Template ID is missing

2 - Invalid template ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Email.Template.Delete

Description

Delete email templates

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Email.Template.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Templates = {string} (required)
Template IDs separated by comma for deleting

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - Template IDs are missing
- 2 - Templates suitable for deleting not found
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

ListIntegration.AddURL

Description

Adds a web service url for list subscription or unsubscription events.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {ListIntegration.AddURL} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer} (required)
ID of subscriber list

Event = {subscription | unsubscription} (required)
Event of the integration

URL = {string} (required)
URL of the web service

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

WebServiceIntegrationID = {integer}
ID of new web service integration url

Error Codes

1 - Subscriber list id is missing

2 - URL is missing

3 - Event type is missing

4 - Invalid subscriber list id

99998 - Authentication failure or session expired

99999 - Not enough privileges

ListIntegration.GetURLs

Description

Retrieves web service integration URLs of a subscriber list.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {ListIntegration.GetURLs} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer} (required)

ID of subscriber list

Event = {subscription | unsubscription} (required)

Event of the integration

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

URLs = {mixed}

Web service integration urls

Error Codes

1 - Subscriber list id is missing

2 - Event type is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

ListIntegration.TestURL

Description

Tests an URL.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {ListIntegration.TestURL} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

URL = {integer} (required)

URL address to be tested

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

... - HTTP error code

99998 - Authentication failure or session expired

99999 - Not enough privileges

ListIntegration.DeleteURLs

Description

Deletes given web service urls.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {ListIntegration.DeleteURLs} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

URLs = {integer} (required)
Comma delimited url ids.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Web service integration url ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

ListIntegration.GenerateSubscriptionFormHTMLCode

Description

Generates subscription form html code.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {ListIntegration.GenerateSubscriptionFormHTMLCode} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

SubscriberListID = {integer} (required)
Subscriber list id.

CustomFields = {string} (required)
Comma delimited custom field ids.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

HTMLCode = {string}
Subscription form html code

Error Codes

1 - Subscriber list id is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

ListIntegration.GenerateUnsubscriptionFormHTMLCode

Description

Generates unsubscription form html code.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {ListIntegration.GenerateUnsubscriptionFormHTMLCode} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer} (required)
Subscriber list id.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code
HTMLCode = {string}
Unsubscription form html code

Error Codes

1 - Subscriber list id is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

List.Create

Description

Creates a new subscriber list

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {List.Create} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListName = {string} (required)
Name of the subscriber list to be created.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ListID = {integer}
ID of the new subscriber list

Error Codes

- 1 - Subscriber list name is missing
- 2 - There is already a subscriber list with given name
- 3 - Allowed list amount exceeded
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

List.Update

Description

Updates subscriber list information.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {List.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallBack = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer} (required)

ID of the subscriber list to be updated.

Name = {string} (required)

name of the subscriber list.

OptInMode = {Single | Double} (required)

Opt-in mode of the subscriber list.

OptInConfirmationEmailID = {integer} (required)

ID number of the email which will be used for sending opt-in confirmation email (required if opt-in mode is set to double)

OptOutScope = {This list | All lists} (required)

Opt-out scope of the subscriber list.

OptOutAddToSuppressionList = {Yes | No} (required)

Set 'Yes' to add unsubscribed email address into the suppression list

SendServiceIntegrationFailedNotification = {true | false} (required)

If set to true, a notification email will be sent to list owner when a web service integration fails.

SubscriptionConfirmationPendingPageURL = {string} (required)

URL of the subscription confirmation pending page. This page is showed only in double opt-in lists.

SubscriptionConfirmedPageURL = {string} (required)

URL of the subscription confirmed page.

HideInSubscriberArea = {true | false} (required)

If set to true, subscriber list will not be shown in subscriber area.

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

- 1 - Subscriber list id is missing
- 2 - Invalid subscriber list id
- 3 - Invalid opt-in mode
- 4 - Invalid opt-out scope
- 5 - Invalid send service integration notification setting
- 6 - Invalid setting
- 7 - Opt in confirmation email id is missing
- 8 - Synchronization fields must be filled in
- 9 - Invalid email address for request processing
- 10 - Invalid add to suppression list on opt-out event value
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Lists.Get

Description

Retrieves subscriber lists of logged in user.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Lists.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallBack = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

OrderField = {field name of subscriber list} (required)

Order field

OrderType = {ASC | DESC} (required)

Order type

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

TotalListCount = {integer}

Total number of subscriber lists

Lists = {mixed}

Returned lists

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

List.Get

Description

Retrieves subscriber list.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {List.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer} (required)

ID of subscriber list to retrieve

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

List = {mixed}

Returned list information

Error Codes

1 - Subscriber list id is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Lists.Delete

Description

Deletes given subscriber lists.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Lists.Delete} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Lists = {string} (required)

Comma delimited subscriber list ids.

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - Subscriber list ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Media.Upload

Description

Uploads media to the media library

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Media.Upload} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

MediaData = {string} (required)

Media file contents encoded with base64

MediaType = {string} (required)

MIME type of the media. Ex: image/gif

MediaSize = {integer} (required)

File size of the media file in bytes

MediaName = {integer} (required)

File name of the media file

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

MediaID = {integer}

ID of the new subscriber list

Error Codes

- 1 - Missing media data
- 2 - Missing media type
- 3 - Missing media size
- 4 - Media file exceeds allowed size
- 5 - Missing media name
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Media.Browse

Description

Returns the list of media available in the media library

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Media.Browse} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

TotalMedia = {integer}
Total number of media files in the library

MediaFiles = {mixed}
List of media files in the library

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Media.Retrieve

Description

Retrieves details of a media item

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Media.Retrieve} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

MediaID = {integer} (required)
Media ID of the media item

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Missing media ID

2 - Invalid media ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Media.Delete

Description

Delete a media item

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Media.Delete} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

MediaID = {integer} (required)
Media ID of the media item

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Missing media ID

2 - Invalid media ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Segment.Create

Description

Create a new segment under a subscriber list

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Segment.Create} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer} (required)
Subscriber list ID to create segment under

SegmentName = {string} (required)
Name of the new segment

SegmentOperator = {and | or} (required)
Match all or any rules

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Missing subscriber list ID

2 - Missing segment name

3 - Missing segment operator

99998 - Authentication failure or session expired

99999 - Not enough privileges

Segment.Update

Description

Updates Segment.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Segment.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SegmentID = {integer} (required)
Subscriber list ID to create segment under

SegmentName = {string} (required)
Name of the new segment

SegmentOperator = {AND | OR} (required)
Match all or any rules

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Missing segment id

2 - Missing segment name

3 - Missing segment operator

4 - Invalid segment id

5 - Invalid segment operator

99998 - Authentication failure or session expired

99999 - Not enough privileges

Segment.Get

Description

Retrieves segments of a subscriber list.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Segment.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

OrderField = {string} (required)

Field name to order rows

OrderType = {ASC | DESC} (required)

Ascending or descending sorting

SubscriberListID = {integer} (required)

Subscriber list ID to return segments

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

ErrorText = {text}

TotalSegments = {integer}

Number of segments returned

Segments = {array}
Returned segment rows

Error Codes

1 - Missing subscriber list id

99998 - Authentication failure or session expired

99999 - Not enough privileges

Segments.Delete

Description

Deletes given segments.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Segments.Delete} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Segments = {string} (required)

Comma delimited Segment ids

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Segment ids are missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Segments.Copy

Description

Copies segments from a selected subscriber list.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Segments.Copy} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SourceListID = {integer} (required)
Copy segments from subscriber list ID

TargetListID = {integer} (required)
Copy segments to subscriber list ID

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Missing source subscriber list id

2 - Missing target subscriber list id

3 - Invalid source subscriber list id

4 - Invalid target subscriber list id

99998 - Authentication failure or session expired

99999 - Not enough privileges

Subscriber.Login

Description

Logs in the subscriber

Request Parameters

Command = {Subscriber.Login} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer} (required)
List ID that stores the subscriber information

MSubscriberID = {integer} (required)
ID of the subscriber (md5)

MEmailAddress = {string} (required)
Email address of the subscriber (md5)

EmailAddress = {string} (required)
Email address to validate

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Missing list ID

2 - Missing subscriber ID

3 - Missing subscriber email address

4 - Missing validation email address

5 - Invalid list ID

6 - Invalid user

7 - Invalid validation email address

8 - Invalid login information

99998 - Authentication failure or session expired

99999 - Not enough privileges

Subscriber.GetLists

Description

Returns subscriber lists of the user

"Important:" This command requires subscriber login session which can be retrieved by running `[[Subscriber.Login]]` API call.

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscriber.GetLists} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

SubscribedLists = {array}
Returns the list of subscriber lists and subscriptions in array

Error Codes

- 1 - Invalid authentication #1
- 2 - Invalid authentication #2
- 3 - Invalid subscriber information

99998 - Authentication failure or session expired

99999 - Not enough privileges

Subscriber.Get

Description

Retrieve subscriber information

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscribers.Get} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailAddress = {string} (required)

Email address of the target subscriber

ListID = {integer} (required)

ID of the list which email address is subscribed to

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

ErrorText = {text}

SubscriberInformation = {array}
Returns the subscriber information

Error Codes

- 1 - Missing email address
- 2 - Missing subscriber list ID
- 3 - Subscriber doesn't exist
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Subscribers.Get

Description

Retrieves subscribers of a subscriber list.

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscribers.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

OrderField = {string} (required)
Name of the field to order based on

OrderType = {ASC | DESC} (required)
Ascending or descending ordering

RecordsFrom = {integer} (required)
Start from (starts from zero)

RecordsPerRequest = {integer} (required)
How many rows to return per page

SearchField = {string} (required)
Subscriber list field to make the search. Leave empty to disable search

SearchKeyword = {string} (required)
The keyword to search in subscriber list database. Leave empty to disable search

SubscriberListID = {integer} (required)
List ID of the subscribers

SubscriberSegment = {mixed} (required)
Target segment ID or one of the following values: Suppressed, Unsubscribed, Soft bounced, Hard bounced, Opt-in pending, Active.

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorText = {text}

Subscribers = {array}
Returns the list of all subscribers in array

TotalSubscribers = {integer}
Returns the total number of subscribers

Error Codes

1 - Missing subscriber list ID

2 - Target segment ID is missing

99998 - Authentication failure or session expired

99999 - Not enough privileges

Subscriber.Subscribe

Description

Subscribes an email address to provided subscriber list or lists

Request Parameters

SessionID = {string} (optional)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscriber.Subscribe} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer} (required)

Target List ID

EmailAddress = {true | false} (required)

Email address to subscribe

CustomFieldX = {array} (required)

Additional information about the subscriber. Replace X with the ID number of the custom field.

IPAddress = {string} (required)

IP address of subscriber

Response Parameters

Success = {true | false}

States if there is an error or not

SubscriberID = {integer}

Subscriber ID of the email address will be returned if the result is success

RedirectURL = {string}
Target URL to redirect user after the process

ErrorCode = {integer}
If there is an error, error's code

ErrorCustomFieldID = {integer}
If there's an error with one of the provided custom fields, custom field ID number will be provided

ErrorCustomFieldTitle = {integer}
If there's an error with one of the provided custom fields, custom field title will be provided

ErrorCustomFieldDescription = {integer}
If there's an error with one of the provided custom fields, description of the error message is provided

Error Codes

- 1 - Target subscriber list ID is missing
- 2 - Email address is missing
- 3 - IP address of subscriber is missing
- 4 - Invalid subscriber list ID
- 5 - Invalid email address
- 6 - One of the provided custom fields is empty. Custom field ID and title is provided as an additional output parameter
- 7 - One of the provided custom field value already exists in the database. Please enter a different value. Custom field ID and title is provided as an additional output parameter
- 8 - One of the provided custom field value failed in validation checking. Custom field ID and title is provided as an additional output parameter
- 9 - Email address already exists in the list
- 10 - Unknown error occurred
- 11 - Invalid user information

99998 - Authentication failure or session expired

99999 - Not enough privileges

Subscriber.OptIn

Description

Perform opt-in confirmation or reject

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscriber.OptIn} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer} (required)

Subscriber List ID

SubscriberID = {integer} (required)

Subscriber ID

Mode = {Confirm | Reject} (required)

Performs opt-in confirmation or reject

Response Parameters

Success = {true | false}

States if there is an error or not

ProcessedMode = {Confirm | Reject}

returns the process completed

RedirectURL = {string}
Target URL to redirect user after the process

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - Subscriber list ID is missing
- 2 - Subscriber ID is missing
- 3 - Opt mode is missing
- 4 - Invalid subscriber list ID
- 5 - Invalid user ID
- 6 - Invalid subscriber ID
- 7 - Invalid opt-in process mode
- 8 - Subscriber already opt-in confirmed
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Subscriber.Unsubscribe

Description

Unsubscribe the subscriber from the list

Request Parameters

SessionID = {string} (optional)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscriber.Unsubscribe} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ListID = {integer} (required)
Subscriber List ID

CampaignID = {integer} (required)
If link is generated for an email campaign, campaign ID should be provided for statistics

EmailID = {integer}
If email ID is provided, the unsubscription statistics will be registered to that email and owner A/B split testing campaign

SubscriberID = {integer} (required)
Subscriber ID or email address must be provided

EmailAddress = {string} (required)
Subscriber ID or email address must be provided

IPAddress = {string} (required)
IP address of the user who has requested to unsubscribe

Preview = {1 | 0} (required)
If set to 1, unsubscription process will be simulated

Response Parameters

Success = {true | false}
States if there is an error or not

RedirectURL = {string}
Target URL to redirect user after the process

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Subscriber list ID is missing

- 2 - IP address is missing
- 3 - Email address or subscriber ID must be provided
- 4 - Invalid subscriber list ID
- 5 - Invalid user information
- 6 - Invalid email address format
- 7 - Invalid subscriber ID or email address. Subscriber information not found in the database
- 8 - Invalid campaign ID
- 9 - Email address already unsubscribed
- 10 - Invalid email ID
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Subscriber.Update

Description

Update subscriber information in the target list

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscriber.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

SubscriberID = {integer} (required)
Target subscriber ID

SubscriberListID = {integer} (required)
Owner subscription list ID

EmailAddress = {string} (required)
Email address

Fields = {array} (required)
Custom field IDs with prefix of 'CustomField'. Ex: Fields[CustomField28]

Access = {subscriber | admin} (required)
User (or subscriber) authentication

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

ErrorCustomFieldID = {integer}
If there's an error with one of the provided custom fields, custom field ID number will be provided

ErrorCustomFieldTitle = {integer}
If there's an error with one of the provided custom fields, custom field title will be provided

ErrorCustomFieldDescription = {integer}
If there's an error with one of the provided custom fields, description of the error message is provided

Error Codes

1 - Missing subscriber ID

2 - Missing subscriber list ID

- 3 - Missing email address
- 4 - Invalid email address
- 5 - Invalid subscriber list ID
- 6 - Invalid subscriber ID
- 7 - Subscriber already exists
- 8 - One of the provided custom fields is empty. Custom field ID and title is provided as an additional output parameter
- 9 - One of the provided custom field value already exists in the database. Please enter a different value. Custom field ID and title is provided as an additional output parameter
- 10 - One of the provided custom field value failed in validation checking. Custom field ID and title is provided as an additional output parameter
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Subscribers.Delete

Description

Deletes subscriber accounts

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Subscribers.Delete} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

SubscriberListID = {integer}

ID number of the target subscriber list

Subscribers = {string}

ID number of subscribers separated by comma (Ex: 1,3,10)

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

ErrorText = {text}

Error Codes

1 - Missing subscriber ID number(s)

2 - Missing subscriber list ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Settings.Update

Description

Update system settings. This API command can be called by administrative privileges

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Settings.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

System_Email_from_Name = {string}

System_Email_from_Email = {string}

System_Email_Replyto_Name = {string}

System_Email_Replyto_Email = {string}

Alert_Recipient_Email = {string}

Report_Abuse_Email = {string}

X_Complaints_To = {string}

S3_Enabled = {true | false}

S3_Access_Id = {string}

S3_Secret_Key = {string}

S3_Bucket = {string}

S3_Medialibrary_Path = {string}

S3_URL = {string}

Media_Upload_Method = {file | database | s3}

Threshold_Soft_Bounce_Detection = {integer}

Fbl_Incoming_Emailaddress = {string}

Bounce_Catchall_Domain = {string}

Load_Balance_Status = {true | false}

Load_Balance_Emails = {integer}

Load_Balance_Sleep = {integer}

Pme_Usage_Type = {Admin | User}

Pme_Default_Account = {string}

Pme_Default_Apikey = {string}

User_Signup_Enabled = {true | false}

User_Signup_Fields = {string}

User_Signup_Reputation = {Trusted | Untrusted}

User_Signup_Language = {string}

Default_Language = {string}

User_Signup_Groupid = {integer}

ID number of the user group which the user will be assigned to. Pass -1 to make this option selectable in user sign up form

User_Signup_Groupids = {string}

ID number of user groups which will be shown in user sign-up form

Default_Themeid = {integer}

Payment_Currency = {string}

Payment_Tax_Percent = {integer}

Payment_Receipt_Email_Subject = {string}

Payment_Receipt_Email_Message = {string}

Admin_Captcha = {true | false}

User_Captcha = {true | false}

Enabled_Plugins = {string}

Send_Method = {SMTP | LocalIMTA | PHPMail | PowerMTA | SaveToDisk}

Send_Method_Localmta_Path = {string}

Send_Method_Powermta_Vmta = {string}

Send_Method_Powermta_Dir = {string}

Send_Method_Savetodisk_Dir = {string}

Send_Method_Smtp_Host = {string}

Send_Method_Smtp_Port = {integer}

Send_Method_Smtp_Secure = { | ssl | tls}

Send_Method_Smtp_Auth = {true | false}

Send_Method_Smtp_Username = {string}

Send_Method_Smtp_Password = {string}

Send_Method_Smtp_Timeout = {integer}

Send_Method_Smtp_Debug = {true | false}

Send_Method_Smtp_Keepalive = {true | false}

Send_Method_Smtp_Msg_Conn = {integer}

Import_Max_Filesize = {integer}

Attachment_Max_Filesize = {integer}

Media_Max_Filesize = {integer}

X_Mailer = {string}

Mail_Engine = {phpmailer | swiftmailer}

Google_Analytics_Source = {string}

Google_Analytics_Medium = {string}

Forward_to_Friend_Header = {string}

Forward_to_Friend_Footer = {string}

Report_Abuse_Friend_Header = {string}

Report_Abuse_Friend_Footer = {string}

User_Signup_Header = {string}

User_Signup_Footer = {string}

Product_Name = {string}

Default_Subscriberarea_Logout_Url = {string}

Pop3_Bounce_Status = {Enabled | Disabled}

Pop3_Bounce_Host = {string}

Pop3_Bounce_Port = {integer}

Pop3_Bounce_Username = {string}

Pop3_Bounce_Password = {string}

Pop3_Fbl_Status= {Enabled | Disabled}

Pop3_Fbl_Host = {string}

Pop3_Fbl_Port = {integer}

Pop3_Fbl_Username = {string}

Pop3_Fbl_Password = {string}

Pop3_Requests_Status = {Enabled | Disabled}

Pop3_Requests_Host = {string}

Pop3_Requests_Port = {integer}

Pop3_Requests_Username = {string}

Pop3_Requests_Password = {string}

Send_Bounce_Notificationemail = {Enabled | Disabled}

Rebranded_Product_Logo = {Enabled | Disabled}

Rebranded_Product_Logo_Type = {string}

PaypalExpressStatus = {Enabled | Disabled}

PaypalExpressBusinessName = {string}

PaypalExpressPurchaseDescription = {string}

PaypalExpressCurrency = {string}

PaypalWPPStatus = {Enabled | Disabled}

Display_Trigger_Send_Engine_Link = {Yes | No}

Default_Optin_Email_Subject = {string}

Default_Optin_Email_Body = {string}

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

EmailSettingsErrorMessage = {string}
If an error occurs on email settings, error message is provided

Error Codes

- 1 - Invalid email address format
- 2 - Invalid field value
- 3 - PreviewMyEmail.com server connection error
- 4 - Invalid PreviewMyEmail.com account information
- 5 - Email settings failed. Check for returned error message
- 6 - POP3 connection failed. Incorrect connection parameters. Returned POP3 error message included in 'EmailSettingsErrorMessage' parameter.

7 - Default opt-in email body does not contain the subscription confirmation link tag (%Link: Confirm%)

99998 - Authentication failure or session expired

99999 - Not enough privileges

Settings.EmailSendingTest

Description

Tests provided email sending settings

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Settings.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

EmailSettingsErrorMessage = {string}
If an error occurs on email settings, error message is provided

Error Codes

1 - Email settings failed. Check for returned error message

2 - Invalid field value

99998 - Authentication failure or session expired

99999 - Not enough privileges

Tag.Create

Description

Create a tag to be used in campaign filtering

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Tag.Create} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Tag = {string} (required)

Name of the tag which is going to be created

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

TagID = {integer}
ID number of the tag

Error Codes

- 1 - Tag name is missing
- 2 - There's an already tag with the same name
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Tags.Get

Description

Returns the list of tags in the user account

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Tags.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

TotalTagCount = {integer}
Number of tag records returned

Tags = {array}
List of tags

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Tag.Update

Description

Update an existing tag

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Tag.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

TagID = {integer} (required)
ID number of the target tag

Tag = {string} (required)
New name of the tag

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Tags.Delete

Description

Delete tags

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Tags.Delete} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Tags = {string} (required)
Tag ID numbers separated with comma

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Tag.AssignToCampaigns

Description

Assigns tag to target campaigns

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Tag.AssignToCampaigns} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

TagID = {integer} (required)
ID number of the tag which is going to be assigned

CampaignIDs = {string} (required)
ID number of campaigns for the tag assignment

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

Tag.UnassignFromCampaigns

Description

Unassign tag from campaigns

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Tag.UnassignFromCampaigns} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

TagID = {integer} (required)
ID number of the tag which is going to be unassigned from campaigns

CampaignIDs = {string} (required)
ID number of campaigns for the tag removal

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges

User.Get

Description

Returns the user information

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {User.Get} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

UserID = {integer} (required)
ID number of the target user account (or email address)

EmailAddress = {string} (required)
Email address of the target user account (or user ID)

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

UserInformation = {array}
User information

Error Codes

1 - User ID is missing

2 - EmailAddress is missing

3 - No user account found

99998 - Authentication failure or session expired

99999 - Not enough privileges

User.Login

Description

Verifies the provided username and password then logs the user in.

Request Parameters

Command = {User.Login} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Username = {string} (required)
Username of the user to be logged in

Password = {string} (required)
Password of the user to be logged in

RememberMe = {string} (required)
If set to any value, login information will be saved in the cookie and login will be skipped next time

Captcha = {string} (required)
If captcha image verification is enabled, direct API login will be disabled

DisableCaptcha = {true | false} (optional)
If this parameter is passed in, it will by-pass the captcha validation

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

SessionID = {string}
SessionID of the logged in user

Error Codes

1 - Username is missing

2 - Password is missing

3 - Invalid login information

4 - Invalid image verification

5 - Image verification failed

99998 - Authentication failure or session expired

99999 - Not enough privileges

User.PasswordRemind

Description

Sends the password reset email.

Request Parameters

Command = {User.PasswordRemind} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

EmailAddress = {string} (required)

Email address of the user to be reminded

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - Email address is missing

2 - Invalid email address

3 - Email address not found in user database

99998 - Authentication failure or session expired

99999 - Not enough privileges

User.PasswordReset

Description

Resets user's password and sends the new password with email.

Request Parameters

Command = {User.PasswordReset} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

UserID = {integer} (required)

ID of the user whose password will be reset (encrypted with md5)

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Error Codes

1 - User id is missing

2 - Invalid user id

99998 - Authentication failure or session expired

99999 - Not enough privileges

Theme.Create

Description

Creates a new theme

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Theme.Create} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

Template = {string} (required)

Template code

ThemeName = {string} (required)

Name of the new theme

ProductName = {string} (required)

Branding the product, name of the product

ThemeSettings = {string} (required)

CSS theme settings.

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

ThemeID = {integer}

Theme ID of the new theme

Error Codes

- 1 - Missing template code
- 2 - Missing theme name
- 3 - Invalid template code
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Theme.Get

Description

Gets the details of a theme

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Settings.Update} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ThemeID = {integer} (required)

Theme ID to get

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Theme = {array}
Theme data

Error Codes

1 - Missing theme ID

2 - Invalid theme ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Theme.Update

Description

Updates a theme

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Settings.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONPCallback = {bool}
Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

ThemeID = {integer} (required)
Theme ID to get

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

1 - Missing template code

2 - Missing theme name

3 - Invalid template code

4 - Missing theme ID

5 - Invalid theme ID

99998 - Authentication failure or session expired

99999 - Not enough privileges

Theme.Delete

Description

Deletes a theme

Request Parameters

SessionID = {string} (required)

Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Theme.Delete} (required)

API Command

ResponseFormat = {JSON | XML} (required)

Response format

JSONPCallback = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSON#JSONP>

Themes = {string} (required)
ID numbers of themes separated by comma

Response Parameters

Success = {true | false}
States if there is an error or not

ErrorCode = {integer}
If there is an error, error's code

Error Codes

- 1 - At least one theme ID must be provided
- 2 - This is the only theme in the system. It can not be deleted
- 99998 - Authentication failure or session expired
- 99999 - Not enough privileges

Themes.Get

Description

Returns the list of themes

Request Parameters

SessionID = {string} (required)
Session ID which is needed to authenticate the access. It's optional and required for desktop applications only

Command = {Settings.Update} (required)
API Command

ResponseFormat = {JSON | XML} (required)
Response format

JSONP_CALLBACK = {bool}

Send this parameter for cross-domain. More info about JSONP can be found here: <http://en.wikipedia.org/wiki/JSONP>

Response Parameters

Success = {true | false}

States if there is an error or not

ErrorCode = {integer}

If there is an error, error's code

Themes = {array}

List of available themes

Error Codes

99998 - Authentication failure or session expired

99999 - Not enough privileges