
line-bot-sdk-python Documentation

Release 1.0.0

RyosukeHasebe

Apr 28, 2017

Contents

1	linebot package	3
1.1	linebot.api module	3
1.2	linebot.webhook module	5
1.3	linebot.exceptions module	7
1.4	linebot.http_client module	8
1.5	linebot.utils module	10
1.6	Subpackages	11
1.6.1	linebot.models package	11
1.6.1.1	linebot.models.base module	11
1.6.1.2	linebot.models.error module	12
1.6.1.3	linebot.models.events module	12
1.6.1.4	linebot.models.imagemap module	16
1.6.1.5	linebot.models.messages module	17
1.6.1.6	linebot.models.responses module	19
1.6.1.7	linebot.models.send_messages module	20
1.6.1.8	linebot.models.sources module	22
1.6.1.9	linebot.models.template module	24
2	Indices and tables	29
	Python Module Index	31

Contents:

linebot.api module

linebot.api module.

```
class linebot.api.LineBotApi (channel_access_token, endpoint=u'https://api.line.me', timeout=5,  
                             http_client=<class 'linebot.http_client.RequestsHttpClient'>)
```

Bases: object

LineBotApi provides interface for LINE messaging API.

`__init__` method.

Parameters

- **channel_access_token** (*str*) – Your channel access token
- **endpoint** (*str*) – (optional) Default is <https://api.line.me>
- **timeout** (*float* | *tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `linebot.http_client.HttpClient.DEFAULT_TIMEOUT`
- **http_client** (*T* <= `linebot.http_client.HttpClient`) – (optional) Default is `linebot.http_client.RequestsHttpClient`

DEFAULT_API_ENDPOINT = `u'https://api.line.me'`

get_message_content (*message_id*, *timeout=None*)

Call get content API.

<https://devdocs.line.me/en/#get-content>

Retrieve image, video, and audio data sent by users.

Parameters

- **message_id** (*str*) – Message ID

- **timeout** (*float | tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

Return type `linebot.models.responses.MessageContent`

Returns MessageContent instance

get_profile (*user_id, timeout=None*)

Call get profile API.

<https://devdocs.line.me/en/#bot-api-get-profile>

Get user profile information.

Parameters

- **user_id** (*str*) – User ID
- **timeout** (*float | tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

Return type `linebot.models.responses.Profile`

Returns Profile instance

leave_group (*group_id, timeout=None*)

Call leave group API.

<https://devdocs.line.me/en/#leave>

Leave a group.

Parameters

- **group_id** (*str*) – Group ID
- **timeout** (*float | tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

leave_room (*room_id, timeout=None*)

Call leave room API.

<https://devdocs.line.me/en/#leave>

Leave a room.

Parameters

- **room_id** (*str*) – Room ID
- **timeout** (*float | tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

multicast (*to, messages, timeout=None*)

Call multicast API.

<https://devdocs.line.me/en/#multicast>

Send messages to multiple users at any time.

Parameters

- **to** (*list[str]*) – IDs of the receivers Max: 150 users

- **messages** ($T \leq \text{linebot.models.send_messages.SendMessage} \mid \text{list}[T \leq \text{linebot.models.send_messages.SendMessage}]$) – Messages. Max: 5
- **timeout** ($\text{float} \mid \text{tuple}(\text{float}, \text{float})$) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

push_message (*to, messages, timeout=None*)

Call push message API.

<https://devdocs.line.me/en/#push-message>

Send messages to users, groups, and rooms at any time.

Parameters

- **to** (*str*) – ID of the receiver
- **messages** ($T \leq \text{linebot.models.send_messages.SendMessage} \mid \text{list}[T \leq \text{linebot.models.send_messages.SendMessage}]$) – Messages. Max: 5
- **timeout** ($\text{float} \mid \text{tuple}(\text{float}, \text{float})$) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

reply_message (*reply_token, messages, timeout=None*)

Call reply message API.

<https://devdocs.line.me/en/#reply-message>

Respond to events from users, groups, and rooms.

Webhooks are used to notify you when an event occurs. For events that you can respond to, a `replyToken` is issued for replying to messages.

Because the `replyToken` becomes invalid after a certain period of time, responses should be sent as soon as a message is received.

Reply tokens can only be used once.

Parameters

- **reply_token** (*str*) – `replyToken` received via webhook
- **messages** ($T \leq \text{linebot.models.send_messages.SendMessage} \mid \text{list}[T \leq \text{linebot.models.send_messages.SendMessage}]$) – Messages. Max: 5
- **timeout** ($\text{float} \mid \text{tuple}(\text{float}, \text{float})$) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.http_client.timeout`

linebot.webhook module

`linebot.http_client` webhook.

class `linebot.webhook.SignatureValidator` (*channel_secret*)

Bases: `object`

Signature validator.

<https://devdocs.line.me/en/#webhook-authentication>

`__init__` method.

Parameters `channel_secret` (*str*) – Channel secret (as text)

validate (*body*, *signature*)

Check signature.

<https://devdocs.line.me/en/#webhook-authentication>

Parameters

- **body** (*str*) – Request body (as text)
- **signature** (*str*) – X-Line-Signature value (as text)

Return type bool

Returns result

class `linebot.webhook.WebhookHandler` (*channel_secret*)

Bases: object

Webhook Handler.

`__init__` method.

Parameters `channel_secret` (*str*) – Channel secret (as text)

add (*event*, *message=None*)

[Decorator] Add handler method.

Parameters

- **event** (T <= `linebot.models.events.Event` class) – Specify a kind of Event which you want to handle
- **message** – (optional) If event is MessageEvent, specify kind of Messages which you want to handle

Type message: T <= `linebot.models.messages.Message` class

Return type func

Returns decorator

default ()

[Decorator] Set default handler method.

Return type func

Returns

handle (*body*, *signature*)

Handle webhook.

Parameters

- **body** (*str*) – Webhook request body (as text)
- **signature** (*str*) – X-Line-Signature value (as text)

class `linebot.webhook.WebhookParser` (*channel_secret*)

Bases: object

Webhook Parser.

`__init__` method.

Parameters `channel_secret` (*str*) – Channel secret (as text)

parse (*body*, *signature*)

Parse webhook request body as text.

Parameters

- **body** (*str*) – Webhook request body (as text)
- **signature** (*str*) – X-Line-Signature value (as text)

Return type list[T <= *linebot.models.events.Event*]

Returns

`linebot.webhook.compare_digest` (*val1*, *val2*)

`compare_digest` function.

If hmac module has `compare_digest` function, use it. Or not, use `linebot.utils.safe_compare_digest`.

Parameters

- **val1** (*str* | *bytes*) – string or bytes for compare
- **val2** (*str* | *bytes*) – string or bytes for compare

Return type bool

Returns result

linebot.exceptions module

`linebot.exceptions` module.

exception `linebot.exceptions.BaseError` (*message=u''*)

Bases: `exceptions.Exception`

Base Exception class.

`__init__` method.

Parameters **message** (*str*) – Human readable message

exception `linebot.exceptions.InvalidSignatureError` (*message=u''*)

Bases: `linebot.exceptions.BaseError`

When Webhook signature does NOT match, this error will be raised.

`__init__` method.

Parameters **message** (*str*) – Human readable message

exception `linebot.exceptions.LineBotApiError` (*status_code*, *error=None*)

Bases: `linebot.exceptions.BaseError`

When LINE Messaging API response error, this error will be raised.

`__init__` method.

Parameters

- **status_code** (*int*) – http status code
- **error** (`linebot.models.error.Error`) – (optional) Error class object.

linebot.http_client module

linebot.http_client module.

class linebot.http_client.**HttpClient** (*timeout=5*)

Bases: object

Abstract Base Classes of HttpClient.

`__init__` method.

Parameters `timeout` (*float | tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `DEFAULT_TIMEOUT`

Return type `T <= HttpResponse`

Returns HttpResponse instance

`DEFAULT_TIMEOUT = 5`

get (*url, headers=None, params=None, stream=False, timeout=None*)

GET request.

Parameters

- `url` (*str*) – Request url
- `headers` (*dict*) – (optional) Request headers
- `params` (*dict*) – (optional) Request query parameter
- `stream` (*bool*) – (optional) get content as stream
- `timeout` (*float | tuple(float, float)*) – (optional), How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.timeout`

Return type `T <= HttpResponse`

Returns HttpResponse instance

post (*url, headers=None, data=None, timeout=None*)

POST request.

Parameters

- `url` (*str*) – Request url
- `headers` (*dict*) – (optional) Request headers
- `data` – (optional) Dictionary, bytes, or file-like object to send in the body
- `timeout` (*float | tuple(float, float)*) – (optional), How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.timeout`

Return type `T <= HttpResponse`

Returns HttpResponse instance

class linebot.http_client.**HttpResponse**

Bases: object

HttpResponse.

content

Get request body as binary.

headers

Get headers.

iter_content (*chunk_size=1024, decode_unicode=False*)

Get request body as iterator content (stream).

Parameters

- **chunk_size** (*int*) –
- **decode_unicode** (*bool*) –

json

Get request body as json-decoded.

status_code

Get status code.

text

Get request body as text-decoded.

class `linebot.http_client.RequestsHttpClient` (*timeout=5*)

Bases: `linebot.http_client.HttpClient`

HttpClient implemented by requests.

`__init__` method.

Parameters **timeout** (*float | tuple(float, float)*) – (optional) How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `DEFAULT_TIMEOUT`

get (*url, headers=None, params=None, stream=False, timeout=None*)

GET request.

Parameters

- **url** (*str*) – Request url
- **headers** (*dict*) – (optional) Request headers
- **params** (*dict*) – (optional) Request query parameter
- **stream** (*bool*) – (optional) get content as stream
- **timeout** (*float | tuple(float, float)*) – (optional), How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.timeout`

Return type `RequestsHttpResponse`

Returns `RequestsHttpResponse` instance

post (*url, headers=None, data=None, timeout=None*)

POST request.

Parameters

- **url** (*str*) – Request url
- **headers** (*dict*) – (optional) Request headers
- **data** – (optional) Dictionary, bytes, or file-like object to send in the body

- **timeout** (*float* | *tuple(float, float)*) – (optional), How long to wait for the server to send data before giving up, as a float, or a (connect timeout, readtimeout) float tuple. Default is `self.timeout`

Return type *RequestsHttpResponse*

Returns *RequestsHttpResponse* instance

class `linebot.http_client.RequestsHttpResponse` (*response*)

Bases: `linebot.http_client.HttpResponse`

HttpResponse implemented by requests lib's response.

`__init__` method.

Parameters `response` – requests lib's response

content

Get request body as binary.

headers

Get headers.

iter_content (*chunk_size=1024, decode_unicode=False*)

Get request body as iterator content (stream).

Parameters

- **chunk_size** (*int*) –
- **decode_unicode** (*bool*) –

json

Get request body as json-decoded.

status_code

Get status code.

text

Get request body as text-decoded.

linebot.utils module

`linebot.http_client` module.

`linebot.utils.safe_compare_digest` (*val1, val2*)

`safe_compare_digest` method.

Parameters

- **val1** (*str* | *bytes*) – string or bytes for compare
- **val2** (*str* | *bytes*) – string or bytes for compare

`linebot.utils.to_camel_case` (*text*)

Convert to camel case.

Parameters `text` (*str*) –

Return type `str`

Returns

`linebot.utils.to_snake_case(text)`

Convert to snake case.

Parameters `text` (*str*) –

Return type `str`

Returns

Subpackages

linebot.models package

linebot.models.base module

linebot.models.base module.

class `linebot.models.base.Base` (***kwargs*)

Bases: `object`

Base class of model.

Suitable for JSON base data.

`__init__` method.

Parameters `kwargs` –

`as_json_dict` ()

Return dictionary from this object.

Returns `dict`

`as_json_string` ()

Return JSON string from this object.

Return type `str`

Returns

static `get_or_new_from_json_dict` (*data*, *cls*)

Helper function.

If *data* is instance of *cls*, return *data*. Else if *data* is instance of `dict`, create instance from `dict`. Else, return `None`.

Parameters

- `data` –
- `cls` –

Return type `object`

Returns

static `get_or_new_from_json_dict_with_types` (*data*, *cls_map*, *type_key=u'type'*)

Helper function.

If *data* is instance of one of *cls*, return *data*. Else if *data* is instance of `dict`, create instance from `dict`. Else, return `None`.

Parameters

- **data** –
- **cls_map** –
- **type_key** –

Return type object

Returns

classmethod `new_from_json_dict` (*data*)

Create a new instance from a dict.

Parameters **data** – JSON dict

Return type

Returns

linebot.models.error module

linebot.models.error module.

class `linebot.models.error.Error` (*message=None, details=None, **kwargs*)

Bases: `linebot.models.base.Base`

Error response of LINE messaging API.

<https://devdocs.line.me/en/#error-response>

`__init__` method.

Parameters

- **message** (*str*) – Summary of the error
- **details** (list[T <= `linebot.models.error.ErrorDetail`]) – ErrorDetail instance list

:type :param kwargs:

class `linebot.models.error.ErrorDetail` (*message=None, property=None, **kwargs*)

Bases: `linebot.models.base.Base`

ErrorDetail response of LINE messaging API.

<https://devdocs.line.me/en/#error-response>

`__init__` method.

<https://devdocs.line.me/en/#error-response>

Parameters

- **message** (*str*) – Details of the error message
- **property** (*str*) – Related property
- **kwargs** –

linebot.models.events module

linebot.models.events module.

class `linebot.models.events.Beacon` (*type=None, hwid=None, dm=None, **kwargs*)
 Bases: `linebot.models.base.Base`

Beacon.

<https://devdocs.line.me/en/#beacon-event>

`__init__` method.

Parameters

- **type** (*str*) – Type of beacon event
- **hwid** (*str*) – Hardware ID of the beacon that was detected
- **dm** (*str*) – Optional. Device message of beacon which is hex string
- **kwargs** –

device_message

Get dm(device_message) as bytearray.

Return type bytearray

Returns

class `linebot.models.events.BeaconEvent` (*timestamp=None, source=None, reply_token=None, beacon=None, **kwargs*)

Bases: `linebot.models.events.Event`

Webhook BeaconEvent.

<https://devdocs.line.me/en/#beacon-event>

Event object for when a user detects a LINE Beacon. You can reply to beacon events.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** ($T \leq \text{linebot.models.sources.Source}$) – Source object
- **reply_token** (*str*) – Reply token
- **beacon** (`linebot.models.events.Beacon`) – Beacon object
- **kwargs** –

class `linebot.models.events.Event` (*timestamp=None, source=None, **kwargs*)

Bases: `linebot.models.base.Base`

Abstract Base Class of Webhook Event.

<https://devdocs.line.me/en/#webhook-event-object>

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** ($T \leq \text{linebot.models.sources.Source}$) – Source object
- **kwargs** –

class `linebot.models.events.FollowEvent` (*timestamp=None, source=None, reply_token=None, **kwargs*)

Bases: `linebot.models.events.Event`

Webhook FollowEvent.

<https://devdocs.line.me/en/#follow-event>

Event object for when your account is added as a friend (or unblocked). You can reply to follow events.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** (T <= `linebot.models.sources.Source`) – Source object
- **reply_token** (*str*) – Reply token
- **kwargs** –

class `linebot.models.events.JoinEvent` (*timestamp=None, source=None, reply_token=None, **kwargs*)

Bases: `linebot.models.events.Event`

Webhook JoinEvent.

<https://devdocs.line.me/en/#join-event>

Event object for when your account joins a group or talk room. You can reply to join events.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** (T <= `linebot.models.sources.Source`) – Source object
- **reply_token** (*str*) – Reply token
- **kwargs** –

class `linebot.models.events.LeaveEvent` (*timestamp=None, source=None, **kwargs*)

Bases: `linebot.models.events.Event`

Webhook LeaveEvent.

<https://devdocs.line.me/en/#leave-event>

Event object for when your account leaves a group.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** (T <= `linebot.models.sources.Source`) – Source object
- **kwargs** –

class `linebot.models.events.MessageEvent` (*timestamp=None, source=None, reply_token=None, message=None, **kwargs*)

Bases: `linebot.models.events.Event`

Webhook MessageEvent.

<https://devdocs.line.me/en/#message-event>

Event object which contains the sent message. The message field contains a message object which corresponds with the message type. You can reply to message events.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** (T <= *linebot.models.sources.Source*) – Source object
- **reply_token** (*str*) – Reply token
- **message** (T <= *linebot.models.messages.Message*) – Message object
- **kwargs** –

class `linebot.models.events.Postback` (*data=None, **kwargs*)

Bases: *linebot.models.base.Base*

Postback.

<https://devdocs.line.me/en/#postback-event>

`__init__` method.

Parameters

- **data** (*str*) –
- **kwargs** –

class `linebot.models.events.PostbackEvent` (*timestamp=None, source=None, reply_token=None, postback=None, **kwargs*)

Bases: *linebot.models.events.Event*

Webhook PostbackEvent.

<https://devdocs.line.me/en/#postback-event>

Event object for when a user performs an action on a template message which initiates a postback. You can reply to postback events.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** (T <= *linebot.models.sources.Source*) – Source object
- **reply_token** (*str*) – Reply token
- **postback** (*linebot.models.events.Postback*) – Postback object
- **kwargs** –

class `linebot.models.events.UnfollowEvent` (*timestamp=None, source=None, **kwargs*)

Bases: *linebot.models.events.Event*

Webhook UnfollowEvent.

<https://devdocs.line.me/en/#unfollow-event>

Event object for when your account is blocked.

`__init__` method.

Parameters

- **timestamp** (*long*) – Time of the event in milliseconds
- **source** (*T* <= *linebot.models.sources.Source*) – Source object
- **kwargs** –

linebot.models.imagemap module

linebot.models.imagemap module.

class `linebot.models.imagemap.BaseSize` (*width=None, height=None, **kwargs*)

Bases: *linebot.models.base.Base*

BaseSize.

<https://devdocs.line.me/en/#imagemap-message>

`__init__` method.

<https://devdocs.line.me/en/#imagemap-message>

Parameters

- **width** (*int*) – Width of base image (set to 1040px)
- **height** (*int*) – Height of base imageset to the height that corresponds to a width of 1040px
- **kwargs** –

class `linebot.models.imagemap.ImagemapAction` (***kwargs*)

Bases: *linebot.models.base.Base*

ImagemapAction.

<https://devdocs.line.me/en/#imagemap-message>

`__init__` method.

Parameters **kwargs** –

class `linebot.models.imagemap.ImagemapArea` (*x=None, y=None, width=None, height=None, **kwargs*)

Bases: *linebot.models.base.Base*

ImagemapArea.

<https://devdocs.line.me/en/#imagemap-area-object>

Defines the size of the full imagemap with the width as 1040px. The top left is used as the origin of the area.

`__init__` method.

Parameters

- **x** (*int*) – Horizontal position of the tappable area
- **y** (*int*) – Vertical position of the tappable area
- **width** (*int*) – Width of the tappable area
- **height** (*int*) – Height of the tappable area
- **kwargs** –

class `linebot.models.imagemap.ImagemapSendMessage` (*base_url=None*, *alt_text=None*,
base_size=None, *actions=None*,
***kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

`ImagemapSendMessage`.

<https://devdocs.line.me/en/#imagemap-message>

Imagemaps are images with one or more links. You can assign one link for the entire image or multiple links which correspond to different regions of the image.

`__init__` method.

Parameters

- **base_url** (*str*) – Base URL of image. HTTPS
- **alt_text** (*str*) – Alternative text
- **base_size** (`linebot.models.imagemap.BaseSize`) – Width and height of base image
- **actions** (`list[T <= linebot.models.imagemap.ImagemapAction]`) – Action when tapped
- **kwargs** –

class `linebot.models.imagemap.MessageImagemapAction` (*text=None*, *area=None*, ***kwargs*)

Bases: `linebot.models.imagemap.ImagemapAction`

`MessageImagemapAction`.

<https://devdocs.line.me/en/#imagemap-message>

`__init__` method.

Parameters

- **text** (*str*) – Message to send
- **area** (`linebot.models.imagemap.ImagemapArea`) – Defined tappable area
- **kwargs** –

class `linebot.models.imagemap.URIIimagemapAction` (*link_uri=None*, *area=None*, ***kwargs*)

Bases: `linebot.models.imagemap.ImagemapAction`

`URIIimagemapAction`.

<https://devdocs.line.me/en/#imagemap-message>

`__init__` method.

Parameters

- **link_uri** (*str*) – Webpage URL
- **area** (`linebot.models.imagemap.ImagemapArea`) – Defined tappable area
- **kwargs** –

linebot.models.messages module

`linebot.models.messages` module.

class `linebot.models.messages.AudioMessage` (*id=None, **kwargs*)
Bases: `linebot.models.messages.Message`

AudioMessage.

<https://devdocs.line.me/en/#audio-message>

Message object which contains the audio content sent from the source. The binary audio data can be retrieved with the Content API.

`__init__` method.

Parameters

- **id** (*str*) – Message ID
- **kwargs** –

class `linebot.models.messages.ImageMessage` (*id=None, **kwargs*)
Bases: `linebot.models.messages.Message`

ImageMessage.

<https://devdocs.line.me/en/#image-message>

Message object which contains the image content sent from the source. The binary image data can be retrieved with the Content API.

`__init__` method.

Parameters

- **id** (*str*) – Message ID
- **kwargs** –

class `linebot.models.messages.LocationMessage` (*id=None, title=None, address=None, latitude=None, longitude=None, **kwargs*)
Bases: `linebot.models.messages.Message`

LocationMessage.

<https://devdocs.line.me/en/#location-message>

`__init__` method.

Parameters

- **id** (*str*) – Message ID
- **title** (*str*) – Title
- **address** (*str*) – Address
- **latitude** (*float*) – Latitude
- **longitude** (*float*) – Longitude
- **kwargs** –

class `linebot.models.messages.Message` (*id=None, **kwargs*)
Bases: `linebot.models.base.Base`

Abstract Base Class of Message.

`__init__` method.

Parameters

- **id** (*str*) – Message ID

- **kwargs** –

class `linebot.models.messages.StickerMessage` (*id=None, package_id=None, sticker_id=None, **kwargs*)

Bases: `linebot.models.messages.Message`

StickerMessage.

<https://devdocs.line.me/en/#sticker-message>

Message object which contains the sticker data sent from the source. For a list of basic LINE stickers and sticker IDs, see sticker list.

`__init__` method.

Parameters

- **id** (*str*) – Message ID
- **package_id** (*str*) – Package ID
- **sticker_id** (*str*) – Sticker ID
- **kwargs** –

class `linebot.models.messages.TextMessage` (*id=None, text=None, **kwargs*)

Bases: `linebot.models.messages.Message`

TextMessage.

<https://devdocs.line.me/en/#text-message>

Message object which contains the text sent from the source.

`__init__` method.

Parameters

- **id** (*str*) – Message ID
- **text** (*str*) – Message text
- **kwargs** –

class `linebot.models.messages.VideoMessage` (*id=None, **kwargs*)

Bases: `linebot.models.messages.Message`

VideoMessage.

<https://devdocs.line.me/en/#video-message>

Message object which contains the video content sent from the source. The binary video data can be retrieved with the Content API.

`__init__` method.

Parameters

- **id** (*str*) – Message ID
- **kwargs** –

linebot.models.responses module

linebot.models.responses module.

class `linebot.models.responses.MessageContent` (*response*)

Bases: `object`

`MessageContent`.

<https://devdocs.line.me/ja/#get-content>

`__init__` method.

Parameters

- **response** (T <= `linebot.http_client.HttpResponse`) – `HttpResponse` object
- **kwargs** –

content

Get content.

If content size is large, should use `iter_content`.

Return type `binary`

content_type

Get Content-type header value.

Return type `str`

Returns content-type header value

iter_content (*chunk_size=1024*)

Get content as iterator (stream).

If content size is large, should use this.

Parameters **chunk_size** – Chunk size

Return type `iterator`

Returns

class `linebot.models.responses.Profile` (*display_name=None, user_id=None, picture_url=None, status_message=None, **kwargs*)

Bases: `linebot.models.base.Base`

`Profile`.

<https://devdocs.line.me/en/#bot-api-get-profile>

`__init__` method.

Parameters

- **display_name** (*str*) – Display name
- **user_id** (*str*) – User ID
- **picture_url** (*str*) – Image URL
- **status_message** (*str*) – Status message
- **kwargs** –

linebot.models.send_messages module

`linebot.models.send_messages` module.

class `linebot.models.send_messages.AudioSendMessage` (*original_content_url=None, duration=None, **kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

AudioSendMessage.

<https://devdocs.line.me/en/#audio>

`__init__` method.

Parameters

- **original_content_url** (*str*) – URL of audio file. HTTPS m4a Less than 1 minute Max 10 MB
- **duration** (*long*) – Length of audio file (milliseconds).
- **kwargs** –

class `linebot.models.send_messages.ImageSendMessage` (*original_content_url=None, preview_image_url=None, **kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

ImageSendMessage.

<https://devdocs.line.me/en/#image>

`__init__` method.

Parameters

- **original_content_url** (*str*) – Image URL. HTTPS JPEG Max: 1024 x 1024 Max: 1 MB
- **preview_image_url** (*str*) – Preview image URL HTTPS JPEG Max: 240 x 240 Max: 1 MB
- **kwargs** –

class `linebot.models.send_messages.LocationSendMessage` (*title=None, address=None, latitude=None, longitude=None, **kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

LocationSendMessage.

<https://devdocs.line.me/en/#location>

`__init__` method.

Parameters

- **title** (*str*) – Title
- **address** (*str*) – Address
- **latitude** (*float*) – Latitude
- **longitude** (*float*) – Longitude
- **kwargs** –

class `linebot.models.send_messages.SendMessage` (***kwargs*)

Bases: `linebot.models.base.Base`

Abstract Base Class of SendMessage.

`__init__` method.

Parameters **kwargs** –

class `linebot.models.send_messages.StickerSendMessage` (*package_id=None, sticker_id=None, **kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

StickerSendMessage.

<https://devdocs.line.me/en/#sticker>

`__init__` method.

Parameters

- **package_id** (*str*) – Package ID
- **sticker_id** (*str*) – Sticker ID
- **kwargs** –

class `linebot.models.send_messages.TextSendMessage` (*text=None, **kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

TextSendMessage.

<https://devdocs.line.me/en/#text>

`__init__` method.

Parameters

- **text** (*str*) – Message text
- **kwargs** –

class `linebot.models.send_messages.VideoSendMessage` (*original_content_url=None, preview_image_url=None, **kwargs*)

Bases: `linebot.models.send_messages.SendMessage`

VideoSendMessage.

<https://devdocs.line.me/en/#video>

`__init__` method.

Parameters

- **original_content_url** (*str*) – URL of video file. HTTPS mp4 Less than 1 minute
Max: 10 MB
- **preview_image_url** (*str*) – URL of preview image. HTTPS JPEG Max: 240 x 240
Max: 1 MB
- **kwargs** –

linebot.models.sources module

linebot.models.sources module.

class `linebot.models.sources.Source` (***kwargs*)

Bases: `linebot.models.base.Base`

Abstract Base Class of Source.

`__init__` method.

Parameters **kwargs** –

sender_id

Helper method.

If SourceUser, return user_id. If SourceGroup, return group_id. If SourceRoom, return room_id.

Return type str

Returns

class `linebot.models.sources.SourceGroup` (*group_id=None, **kwargs*)

Bases: `linebot.models.sources.Source`

SourceGroup.

<https://devdocs.line.me/en/#source-group>

JSON object which contains the source group of the event.

`__init__` method.

Parameters

- **group_id** (*str*) – ID of the source group
- **kwargs** –

sender_id

Helper method.

Alias of group_id.

Return type str

Returns

class `linebot.models.sources.SourceRoom` (*room_id=None, **kwargs*)

Bases: `linebot.models.sources.Source`

SourceRoom.

<https://devdocs.line.me/en/#source-room>

JSON object which contains the source room of the event.

`__init__` method.

Parameters

- **room_id** (*str*) – ID of the source room
- **kwargs** –

sender_id

Helper method.

Alias of room_id.

If SourceUser, return user_id. If SourceGroup, return group_id. If SourceRoom, return room_id.

Return type str

Returns

class `linebot.models.sources.SourceUser` (*user_id=None, **kwargs*)

Bases: `linebot.models.sources.Source`

SourceUser.

<https://devdocs.line.me/en/#source-user>

JSON object which contains the source user of the event.

`__init__` method.

Parameters

- **user_id** (*str*) – ID of the source user
- **kwargs** –

sender_id

Helper method.

Alias of `user_id`.

Return type `str`

Returns

linebot.models.template module

linebot.models.template module.

```
class linebot.models.template.ButtonsTemplate(text=None, title=None, thumbnail_image_url=None, actions=None, **kwargs)
```

Bases: `linebot.models.template.Template`

ButtonsTemplate.

<https://devdocs.line.me/en/#buttons>

Template message with an image, title, text, and multiple action buttons.

`__init__` method.

Parameters

- **text** (*str*) – Message text. Max: 160 characters (no image or title). Max: 60 characters (message with an image or title)
- **title** (*str*) – Title. Max: 40 characters
- **thumbnail_image_url** (*str*) – Image URL. HTTPS JPEG or PNG Aspect ratio: 1:1.51 Max width: 1024px Max: 1 MB
- **actions** (list[T <= `linebot.models.template.TemplateAction`]) – Action when tapped. Max: 4
- **kwargs** –

```
class linebot.models.template.CarouselColumn(text=None, title=None, thumbnail_image_url=None, actions=None, **kwargs)
```

Bases: `linebot.models.base.Base`

CarouselColumn.

<https://devdocs.line.me/en/#column-object>

`__init__` method.

Parameters

- **text** (*str*) – Message text. Max: 120 characters (no image or title) Max: 60 characters (message with an image or title)

- **title** (*str*) – Title. Max: 40 characters
- **thumbnail_image_url** (*str*) – Image URL. HTTPS JPEG or PNG Aspect ratio: 1:1.51 Max width: 1024px Max: 1 MB
- **actions** (list[T <= *linebot.models.template.TemplateAction*]) – Action when tapped. Max: 3
- **kwargs** –

class `linebot.models.template.CarouselTemplate` (*columns=None, **kwargs*)
 Bases: `linebot.models.base.Base`

CarouselTemplate.

<https://devdocs.line.me/en/#carousel>

Template message with multiple columns which can be cycled like a carousel.

`__init__` method.

Parameters

- **columns** (list[T <= *linebot.models.template.CarouselColumn*]) – Array of columns. Max: 5
- **kwargs** –

class `linebot.models.template.ConfirmTemplate` (*text=None, actions=None, **kwargs*)
 Bases: `linebot.models.template.Template`

ConfirmTemplate.

<https://devdocs.line.me/en/#confirm>

Template message with two action buttons.

`__init__` method.

Parameters

- **text** (*str*) – Message text. Max: 240 characters
- **actions** (list[T <= *linebot.models.template.TemplateAction*]) – Action when tapped. Max: 2
- **kwargs** –

class `linebot.models.template.MessageTemplateAction` (*label=None, text=None, **kwargs*)
 Bases: `linebot.models.template.TemplateAction`

MessageTemplateAction.

<https://devdocs.line.me/en/#template-messages>

When this action is tapped, the string in the text field is sent as a message from the user.

`__init__` method.

Parameters

- **label** (*str*) – Label for the action. Max: 20 characters
- **text** (*str*) – Text sent when the action is performed. Max: 300 characters
- **kwargs** –

```
class linebot.models.template.PostbackTemplateAction (label=None, data=None,
                                                    text=None, **kwargs)
```

Bases: `linebot.models.template.TemplateAction`

PostbackTemplateAction.

<https://devdocs.line.me/en/#template-messages>

When this action is tapped, a postback event is returned via webhook with the specified string in the data field.

If you have included the text field, the string in the text field is sent as a message from the user.

`__init__` method.

Parameters

- **label** (*str*) – Label for the action. Max: 20 characters
- **data** (*str*) – String returned via webhook in the `postback.data` property of the postback event. Max: 300 characters
- **text** (*str*) – Text sent when the action is performed. Max: 300 characters
- **kwargs** –

```
class linebot.models.template.Template (**kwargs)
```

Bases: `linebot.models.base.Base`

Abstract Base Class of Template.

`__init__` method.

Parameters **kwargs** –

```
class linebot.models.template.TemplateAction (**kwargs)
```

Bases: `linebot.models.base.Base`

Abstract Base Class of TemplateAction.

`__init__` method.

Parameters **kwargs** –

```
class linebot.models.template.TemplateSendMessage (alt_text=None, template=None,
                                                    **kwargs)
```

Bases: `linebot.models.send_messages.SendMessage`

TemplateSendMessage.

<https://devdocs.line.me/en/#template-messages>

Template messages are messages with predefined layouts which you can customize. There are three types of templates available that can be used to interact with users through your bot.

`__init__` method.

Parameters

- **alt_text** (*str*) – Alternative text for unsupported devices.
- **template** (`T <= linebot.models.template.Template`) – Object with the contents of the template.
- **kwargs** –

```
class linebot.models.template.URITemplateAction (label=None, uri=None, **kwargs)
```

Bases: `linebot.models.template.TemplateAction`

URITemplateAction.

<https://devdocs.line.me/en/#template-messages>

When this action is tapped, the URI specified in the uri field is opened.

`__init__` method.

Parameters

- **label** (*str*) – Label for the action Max: 20 characters
- **uri** (*str*) – URI opened when the action is performed. http, https, tel
- **kwargs** –

CHAPTER 2

Indices and tables

- genindex
- modindex

I

- `linebot.api`, 3
- `linebot.exceptions`, 7
- `linebot.http_client`, 8
- `linebot.models.base`, 11
- `linebot.models.error`, 12
- `linebot.models.events`, 12
- `linebot.models.imagemap`, 16
- `linebot.models.messages`, 17
- `linebot.models.responses`, 19
- `linebot.models.send_messages`, 20
- `linebot.models.sources`, 22
- `linebot.models.template`, 24
- `linebot.utils`, 10
- `linebot.webhook`, 5

A

add() (linebot.webhook.WebhookHandler method), 6
 as_json_dict() (linebot.models.base.Base method), 11
 as_json_string() (linebot.models.base.Base method), 11
 AudioMessage (class in linebot.models.messages), 17
 AudioSendMessage (class in linebot.models.send_messages), 20

B

Base (class in linebot.models.base), 11
 BaseError, 7
 BaseSize (class in linebot.models.imagemap), 16
 Beacon (class in linebot.models.events), 12
 BeaconEvent (class in linebot.models.events), 13
 ButtonsTemplate (class in linebot.models.template), 24

C

CarouselColumn (class in linebot.models.template), 24
 CarouselTemplate (class in linebot.models.template), 25
 compare_digest() (in module linebot.webhook), 7
 ConfirmTemplate (class in linebot.models.template), 25
 content (linebot.http_client.HttpResponse attribute), 8
 content (linebot.http_client.RequestsHttpResponse attribute), 10
 content (linebot.models.responses.MessageContent attribute), 20
 content_type (linebot.models.responses.MessageContent attribute), 20

D

default() (linebot.webhook.WebhookHandler method), 6
 DEFAULT_API_ENDPOINT (linebot.api.LineBotApi attribute), 3
 DEFAULT_TIMEOUT (linebot.http_client.HttpClient attribute), 8
 device_message (linebot.models.events.Beacon attribute), 13

E

Error (class in linebot.models.error), 12

ErrorDetail (class in linebot.models.error), 12
 Event (class in linebot.models.events), 13

F

FollowEvent (class in linebot.models.events), 13

G

get() (linebot.http_client.HttpClient method), 8
 get() (linebot.http_client.RequestsHttpClient method), 9
 get_message_content() (linebot.api.LineBotApi method), 3
 get_or_new_from_json_dict() (linebot.models.base.Base static method), 11
 get_or_new_from_json_dict_with_types() (linebot.models.base.Base static method), 11
 get_profile() (linebot.api.LineBotApi method), 4

H

handle() (linebot.webhook.WebhookHandler method), 6
 headers (linebot.http_client.HttpResponse attribute), 9
 headers (linebot.http_client.RequestsHttpResponse attribute), 10
 HttpClient (class in linebot.http_client), 8
 HttpResponse (class in linebot.http_client), 8

I

ImagemapAction (class in linebot.models.imagemap), 16
 ImagemapArea (class in linebot.models.imagemap), 16
 ImagemapSendMessage (class in linebot.models.imagemap), 16
 ImageMessage (class in linebot.models.messages), 18
 ImageSendMessage (class in linebot.models.send_messages), 21
 InvalidSignatureError, 7
 iter_content() (linebot.http_client.HttpResponse method), 9
 iter_content() (linebot.http_client.RequestsHttpResponse method), 10

iter_content() (linebot.models.responses.MessageContent method), 20

J

JoinEvent (class in linebot.models.events), 14

json (linebot.http_client.HttpResponse attribute), 9

json (linebot.http_client.RequestsHttpResponse attribute), 10

L

leave_group() (linebot.api.LineBotApi method), 4

leave_room() (linebot.api.LineBotApi method), 4

LeaveEvent (class in linebot.models.events), 14

linebot.api (module), 3

linebot.exceptions (module), 7

linebot.http_client (module), 8

linebot.models.base (module), 11

linebot.models.error (module), 12

linebot.models.events (module), 12

linebot.models.imagemap (module), 16

linebot.models.messages (module), 17

linebot.models.responses (module), 19

linebot.models.send_messages (module), 20

linebot.models.sources (module), 22

linebot.models.template (module), 24

linebot.utils (module), 10

linebot.webhook (module), 5

LineBotApi (class in linebot.api), 3

LineBotApiError, 7

LocationMessage (class in linebot.models.messages), 18

LocationSendMessage (class in linebot.models.send_messages), 21

M

Message (class in linebot.models.messages), 18

MessageContent (class in linebot.models.responses), 19

MessageEvent (class in linebot.models.events), 14

MessageImagemapAction (class in linebot.models.imagemap), 17

MessageTemplateAction (class in linebot.models.template), 25

multicast() (linebot.api.LineBotApi method), 4

N

new_from_json_dict() (linebot.models.base.Base class method), 12

P

parse() (linebot.webhook.WebhookParser method), 6

post() (linebot.http_client.HttpClient method), 8

post() (linebot.http_client.RequestsHttpClient method), 9

Postback (class in linebot.models.events), 15

PostbackEvent (class in linebot.models.events), 15

PostbackTemplateAction (class in linebot.models.template), 25

Profile (class in linebot.models.responses), 20

push_message() (linebot.api.LineBotApi method), 5

R

reply_message() (linebot.api.LineBotApi method), 5

RequestsHttpClient (class in linebot.http_client), 9

RequestsHttpResponse (class in linebot.http_client), 10

S

safe_compare_digest() (in module linebot.utils), 10

sender_id (linebot.models.sources.Source attribute), 22

sender_id (linebot.models.sources.SourceGroup attribute), 23

sender_id (linebot.models.sources.SourceRoom attribute), 23

sender_id (linebot.models.sources.SourceUser attribute), 24

SendMessage (class in linebot.models.send_messages), 21

SignatureValidator (class in linebot.webhook), 5

Source (class in linebot.models.sources), 22

SourceGroup (class in linebot.models.sources), 23

SourceRoom (class in linebot.models.sources), 23

SourceUser (class in linebot.models.sources), 23

status_code (linebot.http_client.HttpResponse attribute), 9

status_code (linebot.http_client.RequestsHttpResponse attribute), 10

StickerMessage (class in linebot.models.messages), 19

StickerSendMessage (class in linebot.models.send_messages), 22

T

Template (class in linebot.models.template), 26

TemplateAction (class in linebot.models.template), 26

TemplateSendMessage (class in linebot.models.template), 26

text (linebot.http_client.HttpResponse attribute), 9

text (linebot.http_client.RequestsHttpResponse attribute), 10

TextMessage (class in linebot.models.messages), 19

TextSendMessage (class in linebot.models.send_messages), 22

to_camel_case() (in module linebot.utils), 10

to_snake_case() (in module linebot.utils), 10

U

UnfollowEvent (class in linebot.models.events), 15

URIImagemapAction (class in linebot.models.imagemap), 17

URITemplateAction (class in linebot.models.template), 26

V

validate() (linebot.webhook.SignatureValidator method),
6

VideoMessage (class in linebot.models.messages), 19

VideoSendMessage (class in
linebot.models.send_messages), 22

W

WebhookHandler (class in linebot.webhook), 6

WebhookParser (class in linebot.webhook), 6