simple_email Documentation

Release 0.1.0

Zax Rosenberg

Oct 22, 2018

Contents

simple_email is an easy API for sending e-mails based on the Python smtplib and email standard libraries.

CHAPTER 1

Installation

```
pip install simple_email
```

CHAPTER 2

Quickstart

Easy API for sending e-mails based on smtplib and email.

Example

```
>>> from simple_email import Email, EmailClient
>>>
>>> # Note: EMAIL and EMAIL_AUTH refer to hypothetical environment variables
>>> # NEVER hard-code your auth in your code!
>>> email_client = EmailClient(login=EMAIL, password=EMAIL_AUTH,
                               host='smtp.office365.com', port=587)
>>>
>>> msg = Email(from_addr=email_client.login,
                to_addr="an_email_address@domain.com",
                subject="Sample Subject Line",
                body="Here's the body of the e-mail.",
                cc="another_email_address@domain.com")
>>>
>>> msg.add_attachment(b"Test bytes", "test_file.csv")
>>> msg.add_attachment_from_path(path="/some/path/file.ext")
>>>
>>> email_client.send(msg)
```

CHAPTER 3

Classes

Email(from_addr, to_addr, List], subject, . . .)	Formatted e-mail object to be used with EmailClient.send.
EmailClient(login, password, host, port)	Email client that will be used to send and receive e-mails.

3.1 simple_email.Email

class simple_email.**Email**(*from_addr: str, to_addr: Union[str, List], subject: str, body: str, body_type: str = ’plain’, cc: Union[str, List] = ”, body_encoding: str = ’us-ascii’*)

Formatted e-mail object to be used with EmailClient.send.

Example

```
>>> msg = Email(from_addr=email_client.login,
            to_addr="an_email_address@domain.com",
            subject="Sample Subject Line",
            body="Here's the body of the e-mail.",
            cc="another_email_address@domain.com")
>>>
>>> msg.add_attachment(b"Test bytes", "test_file.csv")
>>> msg.add_attachment_from_path(path="/some/path/file.ext")
```

__init__(*from_addr: str, to_addr: Union[str, List], subject: str, body: str, body_type: str = ’plain’, cc: Union[str, List] = ”, body_encoding: str = ’us-ascii’*)

Parameters

- **from_addr** – E-mail address the Email should come from.
- **to_addr** – E-mail address(es) that will be on the To line of the Email; included in Email recipients.

- **subject** – Email's subject line.
- **body** – Text in the body of the e-mail.
- **body_type** – Sets the type of text in the body. Defaults to 'plain', but can be 'plain' or 'html'.
- **cc** – E-mail address(es) that will be on the Cc line of the Email; included in Email recipients.
- **body_encoding** – The encoding of the text in the body. Defaults to 'us-ascii', but can be 'utf-8', etc.

Methods

`__init__`(from_addr, to_addr, List], subject, . . .)	**param from_addr** E-mail address the Email should come from.
`add_attachment`(data, file_name, maintype, . . .)	Add attachment from bytes object in memory.
`add_attachment_from_path`(path)	Add attachment from path's contents.

3.2 simple_email.EmailClient

class `simple_email.`**`EmailClient`**(*login: str*, *password: str*, *host: str*, *port: int*)

Email client that will be used to send and receive e-mails.

Example

```
>>> # `config` below refers to a hypothetical env file.
>>> email_client = EmailClient(login=config.EMAIL, password=config.EMAIL_AUTH,
                               host='smtp.office365.com', port=587)
```

`__init__`(*login: str*, *password: str*, *host: str*, *port: int*)

Instatiate the credentials for the e-mail client without authenticating.

Parameters

- **login** – The e-mail address used to log in with your e-mail provider. E.g. 'an_email_address@domain.com'
- **password** – The password associated with the login. For security, it is strongly recommended that this parameter be fed from an environment variable and NOT in plain text.
- **host** – The smtp host name. E.g. 'smtp.office365.com' for Outlook365.
- **port** – host's port. E.g. 587 for Outlook365.

Methods

__init__(login, password, host, port)	Instatiate the credentials for the e-mail client without authenticating.
send(msg[, from_addr, to_addrs, …])	Send an Email object as an e-mail.

CHAPTER 4

Functions

parse_email_addr(email)	Strips surrounding angle brackets from an e-mail address.
path_to_bytes(path)	Helper function to get a bytes object from the contents of a file on disk.
path_to_attachment_components	Helper function to get a file's bytes and metadata for use with Email.add_attachment.

4.1 simple_email.main.parse_email_addr

`simple_email.main.`**`parse_email_addr`**(*email: str*) → str

Strips surrounding angle brackets from an e-mail address.

4.2 simple_email.main.path_to_bytes

`simple_email.main.`**`path_to_bytes`**(*path: str*) → bytes

Helper function to get a bytes object from the contents of a file on disk.

Parameters **`path`** – Path to a file to import as a bytes object.

Returns File contents loaded from path as bytes in memory.

4.3 simple_email.main.path_to_attachment_components

`simple_email.main.`**`path_to_attachment_components`**()

Helper function to get a file's bytes and metadata for use with Email.add_attachment.

Parameters **`path`** – Path to a file to import as a bytes object.

Returns Tuple of args needed to create an attachment to an Email.

Python Module Index

Index