

**Maciej Kuczyński<sup>1</sup>, Sebastian Garus<sup>2</sup>**<sup>1</sup> Department of Computer Science, Czestochowa University of Technology, Czestochowa, Poland<sup>2</sup> Department of Mechanics and Fundamentals of Machinery Design, Czestochowa University of Technology, Czestochowa, Poland<sup>1</sup>kuczynskimaciej1@poczta.onet.pl, <sup>2</sup>sebastian.garus@pcz.pl

# Email automation application using Python and SMTP/IMAP protocols

## Introduction

The presented invention is created by one of the students to improve the networking purposes while actively participating in the JoinThe.Space association integrating space industry enthusiasts from all around Europe.

## Motivation

The function performer by the creator is strongly correlated with contacting foreign partners and sending multiple emails at once. Most of them are personalized, requiring changes in headers, salutations or organisation names. In order to ease this task, the email automation app was created using Python and proper mail protocols.

## Technical overview

The app uses the smtplib and imaplib libraries for Python – for outbound and inbound mail respectively. Although it is by default set to the Titan Email host (which is used by the JoinThe.Space members), it allows each user to log in independently.

Next, it requires the user to prepare a CSV or Excel file containing a list of the recipients' email addresses and their headers. Each header is then analysed and processed in order to convert any national characters into HTML numeric codes. Next, the text formatting of the message is prepared using HTML insertion into the Python code. It contains gaps that are filled with data from the previously loaded Excel file.

The application connects with the host servers using SMTP and IMAP protocols and sends a list of emails at once. Each one of them is saved in the Sent catalogue.

Moreover, it allows to add attachments, such as PDF files. The user has to only provide a proper path to the attached file and add it to the attachments list.

In the future, a GUI might be implemented, as well as other host pages will be introduced.

```
def sendMail(FROM, PASSWORD, TO, MESSAGE):  
    SMTP_HOST = "smtp.titan.email"  
    SMTP_PORT = 465  
    SERVER = smtplib.SMTP_SSL(SMTP_HOST, SMTP_PORT)  
    SERVER.connect(SMTP_HOST, SMTP_PORT)  
    SERVER.ehlo()  
    SERVER.login(FROM, PASSWORD)  
    SERVER.sendmail(FROM, TO, MESSAGE)  
    SERVER.quit()
```

## An example function sendMail

<input type="checkbox"/>	eps	JoinThe.Space - n...	Mar 16
<input type="checkbox"/>	jam	JoinThe.Space - n...	Mar 16
<input type="checkbox"/>	support	JoinThe.Space - n...	Mar 16
<input type="checkbox"/>	ao	JoinThe.Space - n...	Mar 16
<input type="checkbox"/>	mikael.enelund	JoinThe.Space - n...	Mar 16
<input type="checkbox"/>	chalmers.ventures	JoinThe.Space - n...	Mar 16
<input type="checkbox"/>	andreas.larsson	JoinThe.Space - n...	Mar 16

## The program's output – a list of sent emails

### Conclusions

The project turned out to be fully functional and managed to increase the speed of email sending several times. Thanks to the application, users do not have to send each message separately and fill in the necessary data manually – instead, it is loaded from a single file and sent at once to all recipients. The project spread interest among other JoinThe.Space users and will be introduced widely soon.

### References

<https://docs.python.org/3/library/smtplib.html>  
<https://docs.python.org/3/library/imaplib.html>