

Email and WhatsApp Automation Script

Instructions

Summary

- Install Python 3.9.10.
- Move project folder '*email_wha_auto*' in the destination folder (e.g. '*Downloads*').
- Enable '*main.py*' script access to Google account (Gmail) of the email sender.
- Edit the configuration file '*email_config.ini*' inside the '*files*' folder (can edit w/ Notepad).
- Type the messages you intend to send via email/WhA in the .docx files in the '*files*' folder.
- Insert in the '*files*' folder the CSV file w/ the list of people to contact. The CSV file '*contacts.csv*' must be of the same template as defined in the '*CSV_template.csv*' file.
- Set to 'y' the flags in the CSV file '*contacts.csv*' ('*email_flag*'/'*wha_flag*') for those contacts that you intend to contact via email/wha, then close the CSV file.
- Access to WhatsApp Web on your Browser with the account you intend to send messages from. Then close the browser window.
- Launch '*main.py*' script via cmd line.

Step 1

Install Python 3.9.10 <https://www.python.org/downloads/release/python-3910/>

Select the installer version that suits your computer hardware specifics.

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		1440acb71471e2394befdb30b1a958d1	25800844	SIG
XZ compressed source tarball	Source release		e754c4b2276750fd5b4785a1b443683a	19154136	SIG
macOS 64-bit Intel-only installer	macOS	for macOS 10.9 and later, deprecated	2714cb9e6241cf7e2f9022714a55d27a	30395760	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	c2393ab11a423d817501b8566ab5da9f	38217233	SIG
Windows embeddable package (32-bit)	Windows		c1d2af96d9f3564f57f35cfc3c1006eb	7671509	SIG
Windows embeddable package (64-bit)	Windows		b8e8bfba8e56edcd654d15e3bdc2e29a	8509821	SIG
Windows help file	Windows		784020441c1a25289483d3d8771a8215	9284044	SIG
Windows installer (32-bit)	Windows		457d648dc8a71b6bc32da30a7805c55b	27767040	SIG
Windows installer (64-bit)	Windows	Recommended	747ac35ae667f4ec1ee3b001e9b7dbc6	28909456	SIG

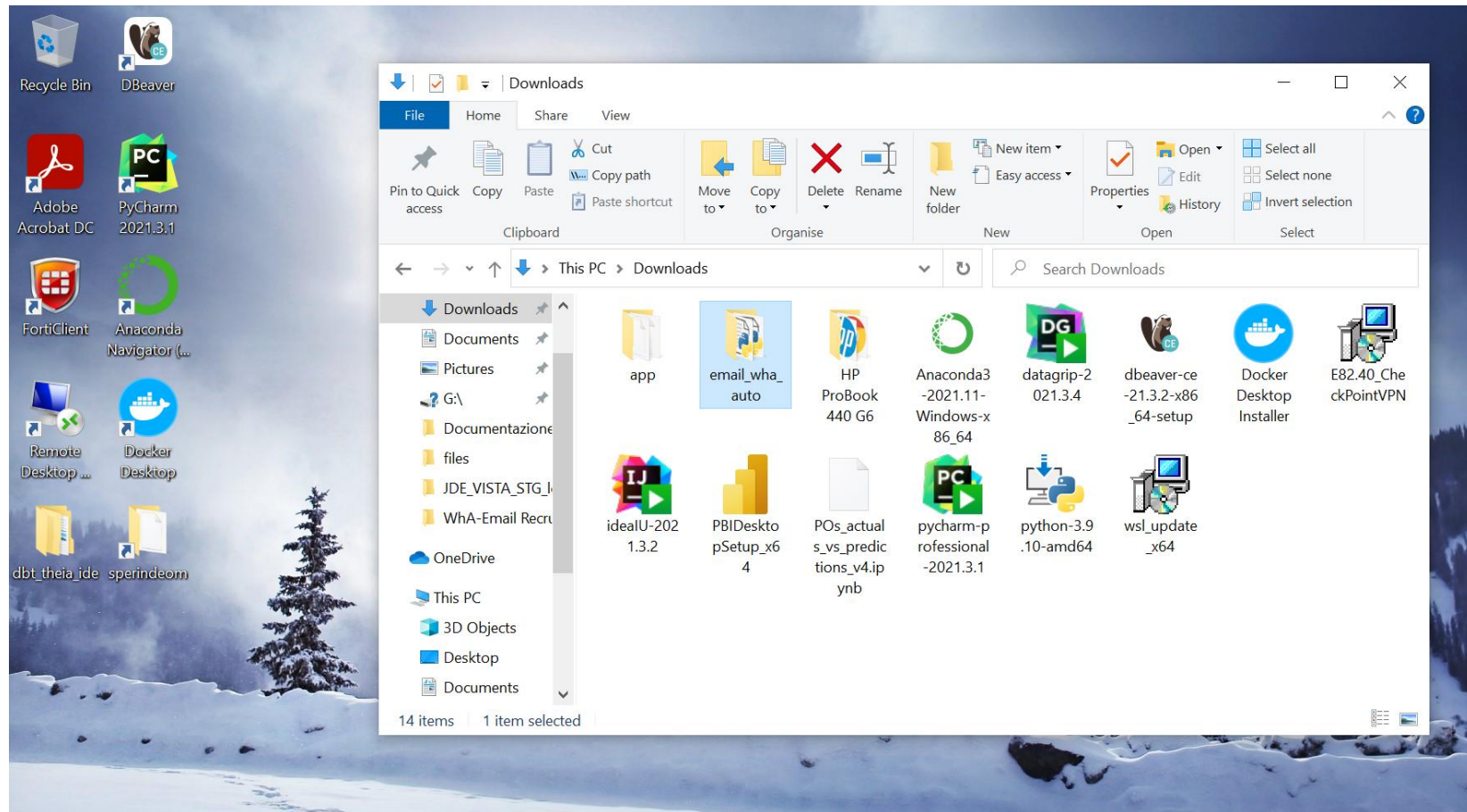
Step 1

Install Python 3.9.10



Step 2

Move project folder '*email_wha_auto*' in the destination folder (e.g. '*Downloads*')



Step 3

Enable '*main.py*' script access to Google account (Gmail) of email sender.

Open Google Chrome -> Settings -> Mange your Google account -> Security -> '*App passwords*' in the '*Signing in to Google*' tab -> Log-in -> App passwords:

← App passwords

App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. [Learn more](#)

You don't have any app passwords.

Select the app and device you want to generate the app password for.

Select app



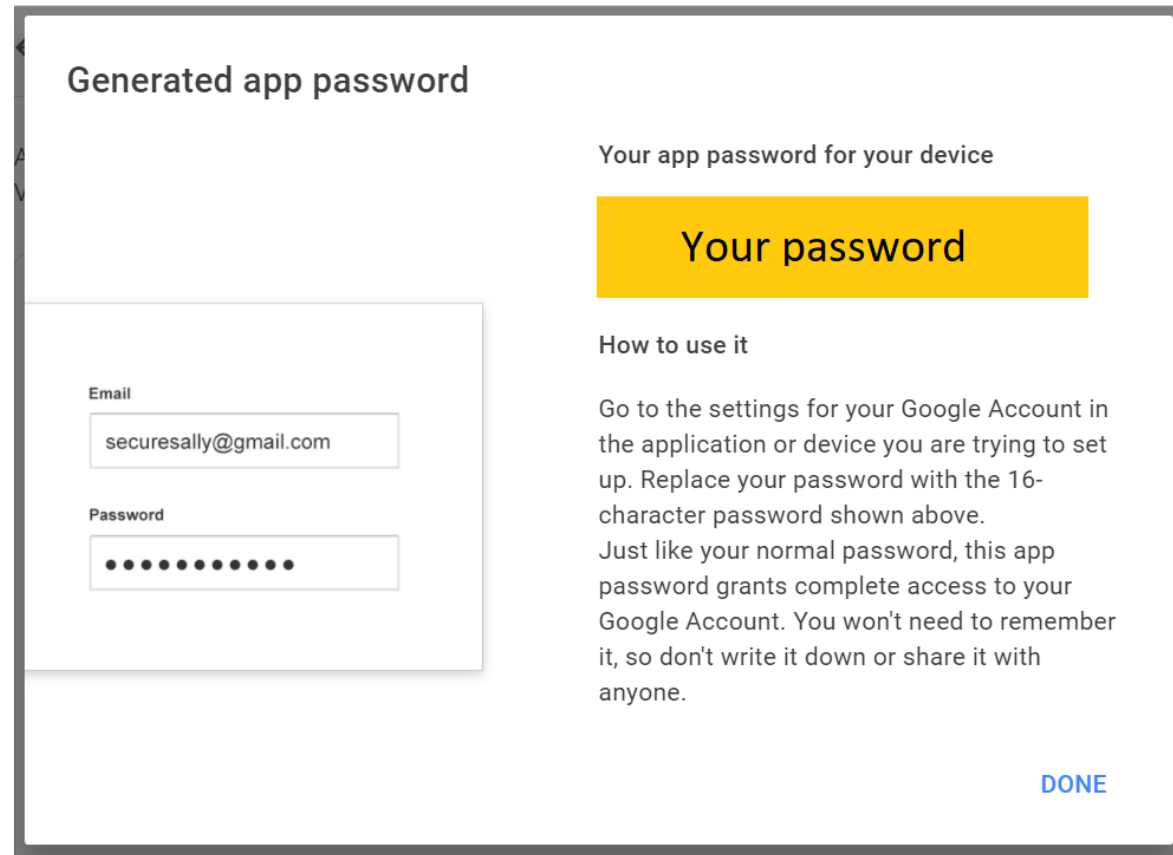
Select device



GENERATE

Step 3

Select '*Other (custom name)*' from the drop-down menu -> '*Select app*' -> type '*main.py*' -> Generate pswd -> Save the pswd that has been created



The screenshot shows a window titled "Generated app password". On the left, there is a form with two fields: "Email" containing "securesally@gmail.com" and "Password" which is masked with dots. On the right, the text "Your app password for your device" is followed by a large yellow box containing the text "Your password". Below this, the section "How to use it" provides instructions: "Go to the settings for your Google Account in the application or device you are trying to set up. Replace your password with the 16-character password shown above. Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone." A blue "DONE" button is located at the bottom right of the window.

Generated app password

Your app password for your device

Your password

How to use it

Go to the settings for your Google Account in the application or device you are trying to set up. Replace your password with the 16-character password shown above. Just like your normal password, this app password grants complete access to your Google Account. You won't need to remember it, so don't write it down or share it with anyone.

DONE

Email

securesally@gmail.com

Password

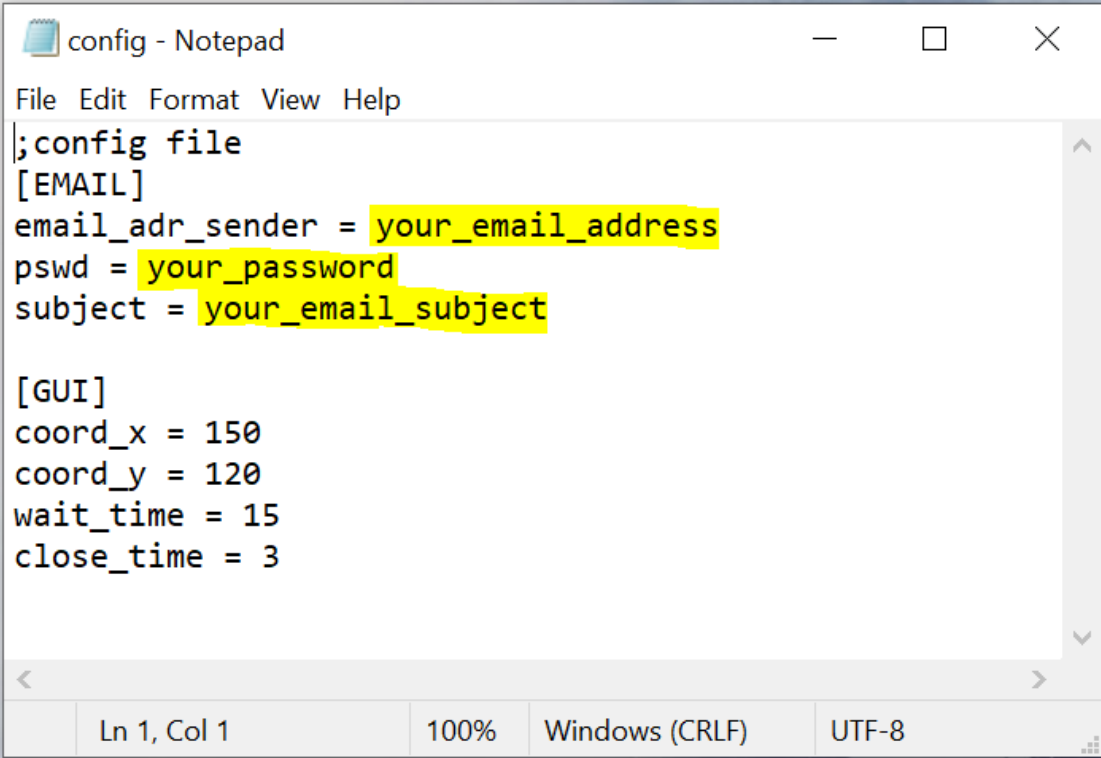
••••••••••••••

Step 4

- Edit the configuration file '*email_config.ini*' inside the '*files*' folder (can edit w/ Notepad).

Insert:

- Sender email address
- Pswd created at previous step
- Email subject for the email you intend to send



```
config - Notepad
File Edit Format View Help
;config file
[EMAIL]
email_adr_sender = your_email_address
pswd = your_password
subject = your_email_subject

[GUI]
coord_x = 150
coord_y = 120
wait_time = 15
close_time = 3

Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

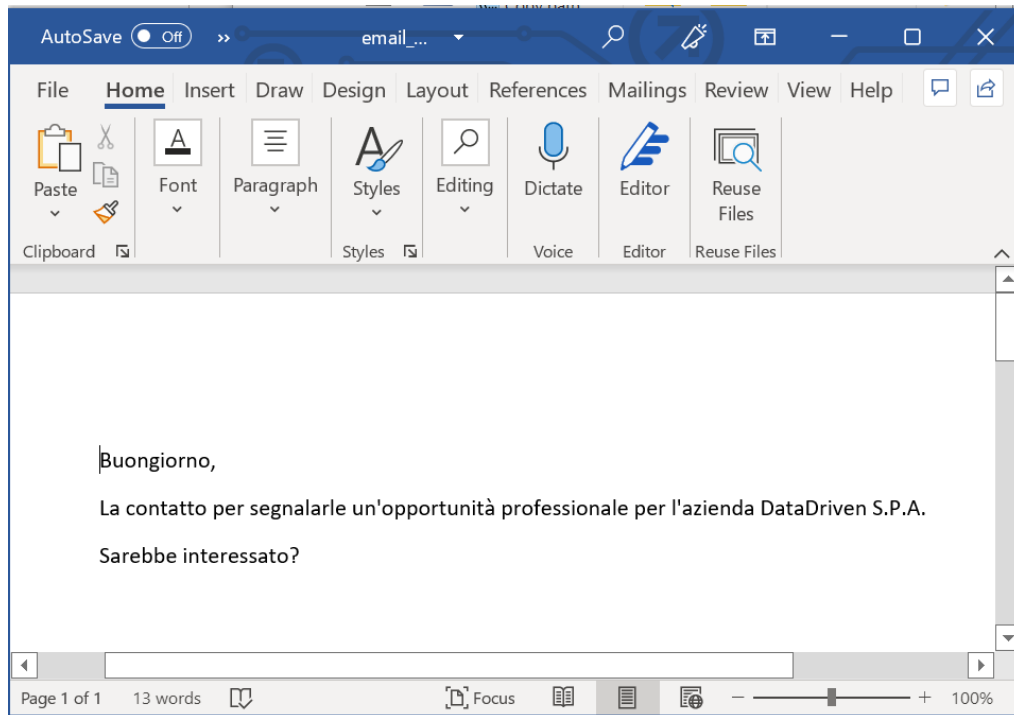

Step 5

Type the messages you intend to send via email/WhA in the .docx files in the 'files' folder.

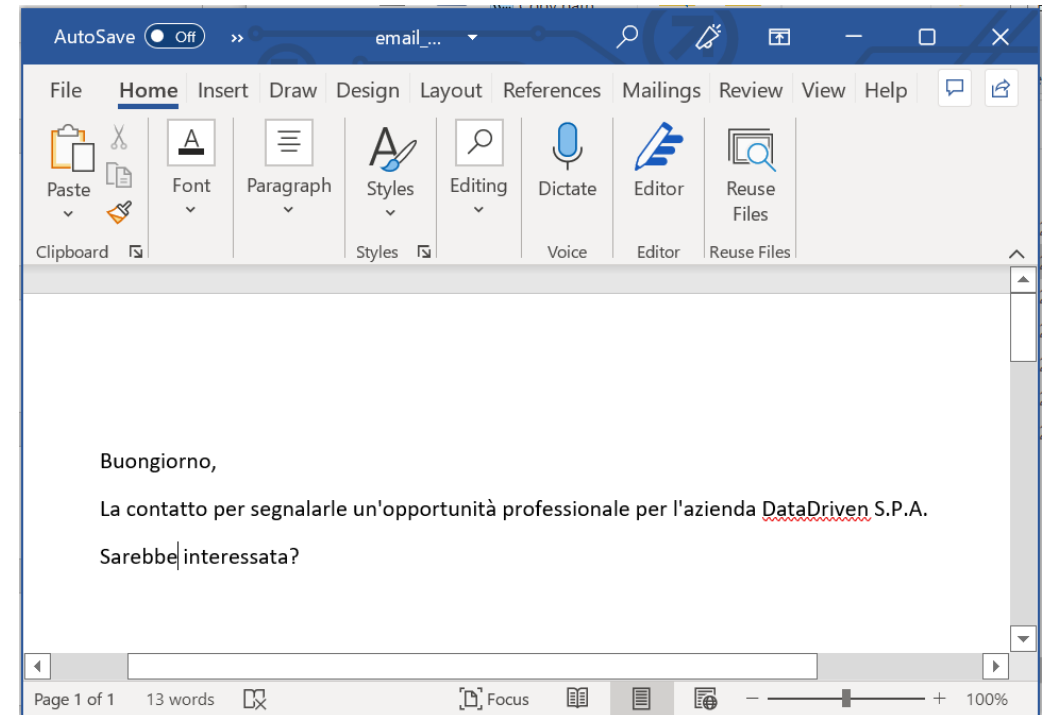
You can insert two different messages for male vs. female contacts in order to customize the msgs according to gender.

E.g.

email_msg_m.docx



email_msg_f.docx



Step 6

Insert in the '*files*' folder the CSV file w/ the list of people to contact. The CSV file '*contacts.csv*' must be of the same template as defined in the '*CSV_template.csv*' file.

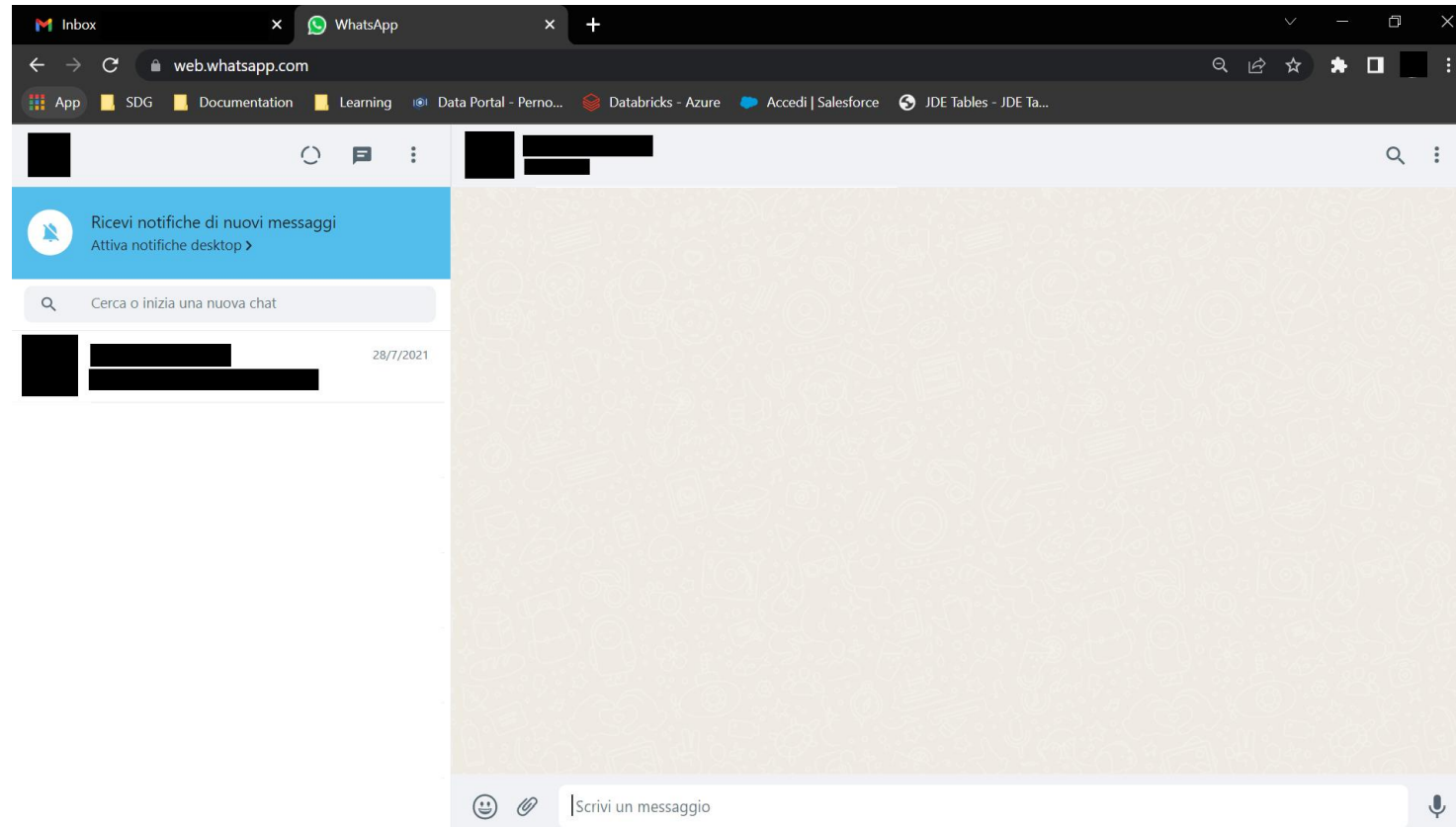
Step 7

Set to 'y' the flags in the CSV file '*contacts.csv*' ('*email_flag*'/'*wha_flag*') for those contacts that you intend to contact via email/wha.

Then **CLOSE THE CSV FILE !!!**

Step 8

Access to WhatsApp Web on your Browser with the account you intend to send messages from. Then close the browser window.



Step 9

Execute '*main.py*' script via cmd line.

1. Open Windows command prompt.
2. Execute the following commands in the order specified:

- `pip install pyautogui`
 - `pip install docx2txt`
 - `cd Downloads\email_wha_auto` (move to destination folder)
 - `python main.py` (execute '*main.py*' script)
- These 2 commands are to be executed only the 1st time script is executed

N.b. In the example shown above the destination folder of the project is assumed to be the '*Downloads*' folder.

Once the run of the '*main.py*' ends, some fields in the CSV file '*contacts.csv*' will be updated.

More specifically these fields are:

- '*email_flag*' e '*wha_flag*' are set as '*n*'.
- '*email_date*' and '*wha_date*' are updated with the date of execution of the script.

Gentle Reminders

For the script to be executed correctly for each row in the '*contacts.csv*' file:

- The field '*gender*' must be set to either '*male*' or '*female*'
Otherwise the script execution fails-
- Make sure that phone number and email address of the candidate are correct.
- If you'd like a given contact to be contacted while executing the script, set the '*email_flag*' and/or '*wha_flag*' to 'y', otherwise if you want a given contact to be skipped and not be contacted set them to 'n'.

N.b. The max no. of emails you can send from a given account per day is **100**