
ClointFusion

Jan 15, 2022

Contents

1	Description	3
1.1	What is ClointFusion?	3
2	Release Notes	5
2.1	Installation	5
3	Windows :	7
4	Ubuntu :	9
4.1	Importing	9
5	Windows :	11
6	Ubuntu :	13
6.1	Features	13
7	WhatsApp Bot - Send bulk WhatsApp messages	15
7.1	ClointFusion in Action	15
8	Now access more than 100 functions (hit ctrl+space in your IDE)	17
9	4 functions on Mouse Operations:	19
10	6 functions on Window Operations (works only in Windows OS):	21
11	8 functions on Folder Operations:	23
12	28 functions on Excel Operations:	25
13	3 functions on Keyboard Operations:	29
14	5 functions on Screen-scraping Operations:	31
15	11 functions on Browser Operations:	33
16	4 functions on Alert Messages:	35
17	3 functions on String Operations:	37

18	Some of miscellaneous functions related to emoji, capture photo, flash (pop-up) messages etc:	39
18.1	ClointFusion's function works in different modes:	39
19	ClointFusion's Semi Automatic Mode	41
19.1	BOTS made out of ClointFusion	41
19.2	We love your contribution	41
20	Invitation to our Monthly Branded Hackathon	43
21	Date with ClointFusion	45
22	Acknowledgements	47
23	Credits	49
23.1	ReadMe File Maintainer	49
24	Need help in Building BOTS?	51



CHAPTER 1

Description

Cloint India Pvt. Ltd - Python functions for Robotic Process Automation shortly RPA.

1.1 What is ClointFusion?

ClointFusion is an Indian firm based in Vadodara, Gujarat. ClointFusion is a Python-based RPA platform for developing Software BOTs. Using AI, we're working on Common Man's RPA.

1.1.1 Check out Project Status

CHAPTER 2

Release Notes

[Click here for Release Notes](#)

2.1 Installation

ClointFusion is now supported on Windows / Ubuntu / macOS* !

CHAPTER 3

Windows :

Windows users can download EXE pre-loaded with Python 3.9 and ClointFusion package: Windows EXE

OR

- ClointFusion is compatible with both Windows 10 and Windows 11.
- Installing on a Windows PC is a breeze.
- Make certain that Python 3.8 or Python 3.9 is installed.
- Then, from the command prompt, execute the following command.

```
pip install -U ClointFusion
```


CHAPTER 4

Ubuntu :

- Clointfusion requires sudo rights to install on Ubuntu.
- Additional Linux packages must be installed before Clointfusion can be installed.
- Make certain that Python 3.8 or Python 3.9 is installed.
- Then, from the command prompt, execute the following command.

```
sudo apt-get install python3-tk python3-dev  
sudo pip3 install ClointFusion
```

4.1 Importing

ClointFusion can be accessed using one of two methods.

CHAPTER 5

Windows :

- **Terminal :** Opens a Python interpreter with “import ClointFusion as cf ” pre-loaded

```
cf_py
```

- **Code Editor or IDE :** Import ClointFusion first, and then run the file in Python.

```
# cf_bot.py  
  
import ClointFusion as cf  
  
cf.browser_activate()
```

```
python cf_bot.py
```


Ubuntu :

- **Terminal :** Opens a Python interpreter with the “import ClointFusion as cf” pre-loaded and the required sudo privileges.

```
sudo cf_py
```

- **Code Editor or IDE :** Run the file with sudo permissions.

```
# cf_bot.py  
  
import ClointFusion as cf  
  
cf.browser_activate()
```

```
sudo python3 cf_bot.py
```

6.1 Features

ClointFusion’s Voice-Guided, Fully Automated Self-Test.

When you import ClointFusion for the first time, or upgrade to a new version, you’ll be prompted with the “ClointFusion’s Automated Self-Test” which highlights all of ClointFusion’s 100+ features in action on your computer while also confirming ClointFusion’s compatibility with your PC’s settings and configurations. Once you have successfully completed the self-test, you will receive an email with a self-test report.

Below is the speed up version of self-test.

[Click here to watch the Self-Test in Action.](#)

- **DOST : Your friend in automation || Build RPA Bots without Code**

DOST is an interactive Blockly based no-code BOT Builder platform built and optimized for ClointFusion-based BOT building. We feel that automation is important for people other than programmers. Using DOST, even a common man can create a BOT in minutes.

Advantages of DOST

- Easy to Use.
- Build BOT in minutes.
- No prior Programming knowledge needed.

6.1.1 Launch DOST client

Windows

Open your favorite browser and go to <https://dost.clointfusion.com> and start building bots.

Note : Make sure ClointFusion Tray is present or open terminal and type `cf_tray` to activate ClointFusion Tray menu.

Ubuntu

Open your favorite terminal and type `sudo dost` and then type `python3 dost.py`.

- Want to change the chrome profile ?
 - Use `python3 dost.py "Profile 1"`

Build BOT with DOST : [DOST Website](#)

BOL : Your automation voice based assistant

BOL is voice based automation assistant designed to execute BOTs build out of ClointFusion without any human computer interaction.

Usage of BOL

Open your favorite terminal and type `bol` or `sudo bol` for ubuntu users. Within a moment, a personalized Virtual Assistant will be at your service.

Note: bol is currently in development stage. More functionalities are yet to be added.

WORK - The Work Hour Monitor

WORK is an intelligent application that detects each and every work you do in your PC and displays a detailed work report.

Usage of WORK

Open your favorite terminal and type `cf_work`. A detailed work report will be displayed.

Note: All the information that is being collected by “WORK“ is stored in a securely maintained database in your system.

WhatsApp Bot - Send bulk WhatsApp messages

ClointFusion's "WhatsApp Bot" is an automated utility tool that allows you to send many customized messages to your contacts at once.

Usage of WhatsApp Bot:

Open your favorite terminal and type `cf_wm`, and give path of the excel, or `cf_wm -e excel_path.xlsx`

[Click here to watch the WhatsApp Bot in Action.](#)

7.1 ClointFusion in Action

Now access more than 100 functions (hit ctrl+space in your IDE)

***TIP:** You can find and inspect all of ClointFusion's functions using only one function i.e., “find()”. Just pass the partial name of the function.*

```
cf.find("sort")
```

```
cf.find("gui")
```


CHAPTER 9

4 functions on Mouse Operations:

Function	Accepted Parameters	Description
cf.mouse_click()	x=" " , y=" " , left_or_right="left", no_of_clicks=1	Clicks at the given X Y Co-ordinates on the screen using single / double / triple click(s). Optionally copies selected data to clipboard (works for double / triple clicks)
cf.mouse_move()	x=" " , y=" "	Moves the cursor to the given X Y Co-ordinates
cf.mouse_drag_from_to()	x1=" " , y1=" " , x2=" " , y2=" " , delay=0.5	Clicks and drags from X1 Y1 co-ordinates to X2 Y2 Co-ordinates on the screen
cf.mouse_search_snip_image()	image=" " , x=" " , y=" "	Searches the given image on the screen and returns its center of X Y co-ordinates.

CHAPTER 10

6 functions on Window Operations (works only in Windows OS):

Function	Accepted Parameters	Description
cf.window_show_desktop()	None	Minimizes all the applications and shows Desktop.
cf.window_get_all_opened_titles_windows()	window_title=” “	Gives the title of all the existing (open) windows.
cf.window_activate_and_maximize_windows()	windowName=” “	Activates and maximizes the desired window.
cf.window_minimize_windows()	windowName=” “	Activates and minimizes the desired window.
cf.window_close_windows()	windowName=” “	Close the desired window.
cf.launch_any_exe_bat_application()	pathOfExeFile=” “	Launches any exe or batch file or excel file etc.

CHAPTER 11

8 functions on Folder Operations:

Function	Accepted Parameters	Description
cf.folder_read_text_file()	file_path=" "	Reads from a given text file and returns entire contents as a single list
cf.folder_write_text_file()	file_path=" ", contents=" "	Writes given contents to a text file
cf.folder_create()	strFolderPath=" "	When you are making leaf directory, if any intermediate-level directory is missing, folder_create() method creates them.
cf.folder_create_text_file()	strFolderPath=" ", txtFileName=" "	Creates text file in the given path.
cf.folder_get_all_files_in_folder()	strFolderPath=" ", extension='all'	Get all the files of the given folder in a list.
cf.folder_delete_all_files()	filePathOfTheFolder=" ", file_extension_without_dot="all"	Deletes all the files of the given folder
cf.file_rename()	old_file_path="", new_file_name="", ext=False	Renames the given file name to new file name with same extension.
cf.file_get_json_details()	path_of_json_file="", section=" "	Returns all the details of the given section in a dictionary

CHAPTER 12

28 functions on Excel Operations:

Function	Accepted Parameters	Description
cf.excel_get_all_sheets()		Gives you all names of the sheets in the given excel sheet.
cf.excel_create_excel_file(folder= "Folder()", excelFileName= " ", sheet_name= "Sheet1")		Creates an excel file in the desired folder with desired filename
cf.excel_if_value_exists(path= " ", sheet_name= "Sheet1", header=0, usecols= " ", value= " ")		Check if a given value exists in given excel. Returns True / False
cf.excel_create_excel_file(fullPathToTheFile= " ", fileName= " ", sheet_name= "Sheet1")		Create a Excel file in fullPathToTheFile with filename.
cf.excel_copy_range(path= " ", sheet_name= "Sheet1", startCol=0, startRow=0, endCol=0, endRow=0, copiedData= " ")		Pastes the copied data in specific range of the given excel sheet.
cf.excel_get_row_col_count(path= " ", sheet_name= "Sheet1", header=0)		Gets the row and column count of the provided excel sheet.
cf.excel_copy_range(path= " ", sheet_name= "Sheet1", startCol=0, startRow=0, endCol=0, endRow=0)		Copies the specific range from the provided excel sheet and returns copied data as a list
cf.excel_split_excel(path= " ", sheet_name= "Sheet1", header=0, columnName= " ")		Splits the excel file by Column Name
cf.excel_split_excel(path= " ", sheet_name= "Sheet1", rowSplitLimit= " ", outputFolderPath= " ", outputTemplateFileName= "Split")		Splits the excel file as per given row limit
cf.excel_merge_all_files(input_folder_path= " ", output_folder_path= " ")		Merges all the excel files in the given folder
cf.excel_drop_column(path= " ", sheet_name= "Sheet1", header=0, columnsToBeDropped = " ")		Drops the desired column from the given excel file
cf.excel_sort_columns(path= " ", sheet_name= "Sheet1", header=0, firstColumnToBeSorted=None, secondColumnToBeSorted=None, thirdColumnToBeSorted=None, firstColumnSortType=True, secondColumnSortType=True, thirdColumnSortType=True, view_excel=False)		A function which takes excel full path to excel and column names on which sort is to be performed
cf.excel_clear_excel(path= " ", sheet_name= "Sheet1", header=0)		Clears the contents of given excel files keeping header row intact
cf.excel_set_single_cell(path= " ", sheet_name= "Sheet1", header=0, columnName= " ", cellNumber=0, setText= " ")		Writes the given text to the desired column/cell number for the
26	Chapter 12. 28 functions on Excel Operations:	
cf.excel_get_single_cell(path= " ", sheet_name= "Sheet1", header=0, columnName= " ", cellNumber=0)		Gets the text from the desired column/cell number of the given excel file

CHAPTER 13

3 functions on Keyboard Operations:

Function	Accepted Parameters	Description
cf.key_hit_enter()	write_to_window=" "	Enter key will be pressed once.
cf.key_press()	key_1=", key_2=", key_3=", write_to_window=" "	Emulates the given keystrokes.
cf.key_write_enter()	text_to_write=" ", write_to_window=" ", delay_after_typing=1, key="e"	Writes/Types the given text and press enter (by default) or tab key.

5 functions on Screen-scraping Operations:

Function	Accepted Parameters	Description
cf.scrape_save_content_to_notepad()	ToSaveTheNotepad="", switch_to_window="X=0, Y=0"	Copy pastes all the available text on the screen to notepad and saves it.
cf.scrape_get_content_highlight_text_copy_paste()		Gets the focus on the screen by searching given text using ctrl+f and performs copy/paste of all data. Useful in Citrix applications. This is useful in Citrix applications
cf.screen_clear_search_highlight_text()	delay=0.2	Clears previously found text (ctrl+f highlight)
cf.search_highlight_tab_and_enter()	tab_and_enter="", hitEnterKey="Yes", shift_tab='No'	Searches for a text on screen using ctrl+f and hits enter. This function is useful in Citrix environment.
cf.find_text_on_screen()	searchText="", delay=0.1, occurrence=1, isSearchToBeCleared=False	Clears previous search and finds the provided text on screen.

CHAPTER 15

11 functions on Browser Operations:

CHAPTER 16

4 functions on Alert Messages:

Function	Accepted Parameters	Description
cf.message_countdown(Message)	Message (ClintFunction in (seconds)", start_value=5	Function to show count-down timer. Default is 5 seconds.
cf.message_popup(Msg)	Msg=" ", delay=3	Specified message will popup on the screen for a specified duration of time.
cf.message_flash(Msg)	Msg=" ", delay=3	Specified msg will popup for a specified duration of time with OK button.
cf.message_toast(message,website_url=" ", file_folder_path=" ")	message,website_url=" ", file_folder_path=" "	Function for displaying Windows 10 Toast Notifications. Pass website URL OR file / folder path that needs to be opened when user clicks on the toast notification.

CHAPTER 17

3 functions on String Operations:

Function	Accepted Parameters	Description
cf.string_remove_special_characters()	inputStr=" "	Removes all the special character.
cf.string_extract_only_alphabets()	inputString=" "	Returns only alphabets from given input string
cf.string_extract_only_numbers()	inputString=" "	Returns only numbers from given input string

CHAPTER 18

Some of miscellaneous functions related to emoji, capture photo, flash (pop-up) messages etc:

Function	Accepted Parameters	Description
cf.clear_screen()	None	Clears Python Interpreter Terminal Window Screen
cf.print_with_magic_color()	Msg:str=" ", magic:bool=False	Function to color and format terminal output
cf.show_emoji()	strInput=" "	Function which prints Emojis
cf.download_this_file()	url=" "	Downloads a given url file to BOT output folder or Browser's Download folder
cf.pause_program()	seconds="5"	Stops the program for given seconds

18.1 ClointFusion's function works in different modes:

ClointFusion's Semi Automatic Mode

1. If you pass all the required parameters, function works silently. So, this is expert (Non-GUI) mode. This mode gives you more control over the function's parameters.
2. If you do not pass any parameter, GUI would pop-up asking you the required parameters. Next time, when you run the BOT, based upon your configuration, which you get to choose at the beginning of BOT run:
 - If `Semi-Automatic` mode is OFF, GUI would pop-up again, showing you the previous entries, allowing you to modify the parameters.
 - If `Semi-Automatic` mode in ON, BOT works silently taking your previous GUI entries.
 - Toggle `Semi-Automatic` mode by using the following command

```
cf.ON_semi_automatic_mode    # To turn ON semi automatic mode  
cf.OFF_semi_automatic_mode   # To turn OFF semi automatic mode
```

3. GUI Mode is for beginners. Anytime, if you are not getting how to use the function, just call an empty function (without parameters) and GUI would pop-up asking you for required parameters.

19.1 BOTS made out of ClointFusion

19.1.1 Outlook Email BOT implemented using ClointFusion

19.2 We love your contribution

Contribute to us by giving a star, writing articles on ClointFusion, giving comments, reporting bugs, bug fixes, feature enhancements, adding documentation, and many other ways.

CHAPTER 20

Invitation to our Monthly Branded Hackathon

We also invite everyone to take part in our monthly branded event, the `ClointFusion` Hackathon, and stand a chance to work with us.

Checkout our Hackathon Website for more details here: [ClointFusion Hackathon](#)

CHAPTER 21

Date with ClointFusion

This an initiative for fast track entry into our growing workforce. For more details, please visit: [Date with ClointFusion](#)

CHAPTER 22

Acknowledgements

We sincerely thanks to all it's dependent packages for the great contribution, which made `ClointFusion` possible!
Please find all the dependencies [here](#)

CHAPTER 23

Credits

23.1 ReadMe File Maintainer

CHAPTER 24

Need help in Building BOTS?

Write us at ClointFusion@cloint.com