# Files Archiving

**Problem Statement:**

When drive space gets low then there is a need to archive sourcing files and deleting the raw files which is a time taking process if being done manually.

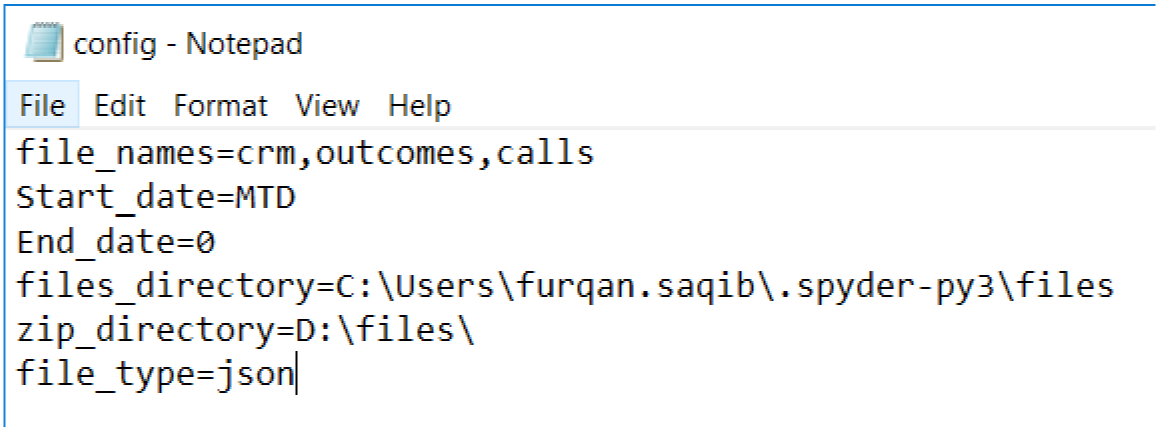
**Purpose of this utility:**

The purpose of this utility is to automate the process of archiving files on all the clients where this activity is being performed manually on monthly basis.

**How to use this Utility:**

Just like other python utilities you just need to fill the configuration file according to your client. Please follow the same

**Configuration File:**



```
file_names=crm,outcomes,calls
Start_date=MTD
End_date=0
files_directory=C:\Users\furqan.saqib\.spyder-py3\files
zip_directory=D:\files\
file_type=json
```

- **File_names**: Here you will give the name of files. i.e if file format is A&G_outcomes_2019-11-01". Since the code extracts the keyword from file name you can give just "outcomes" in such a way it will be unique from the other keywords for the other file names you will provide.
- **Start_date**: if you give MTD then it will pick the first date of the current month. ELSE, you can give any number and it will subtract that number from the current day. i.e 1 means current date-1.
- **End_date**: Give any number and will subtract that number from the current day. i.e 1 means current date-1.
- **Files_directory**: Provide here the path where your sourcing files (that you want to archive) are present.
- **zip_directory**: Provide here the path where you want your zip files to be created.
- **file_type:** sourcing file types i.e CSV, JSON, TXT etc.

Note: Please un-comment the line# 107 in data_archiving.py file after testing it on your client. After this the raw files will start getting deleted.

**Code Working**:

The code will check the directory where your files are present. It will sort the file names in descending order length wise so that file names such as CRM & CRM2 wont get mixed. It will calculate the start and end date being provided by you. Then, it will check the files and for each file if the LAST_MODIFIED_DATE lies with in the date range, it will take all of them and make 1 zip file. This process gets carried on for every file name provided by you.  So, it means if you have provided 5 file names then 5 zip files will be created with in the given date range.

data_archiving.py

config.txt