**Web Harvester Auto Mail Alert**

**1. Overview**

The script is designed to automate the process of updating an Excel file with the latest delivery dates, calculating the next expected delivery dates based on frequency, and sending email alerts if the next delivery date is overdue

**2. Modules and Libraries**

* os: To interact with the file system.
* pandas: To read and manipulate Excel files.
* datetime: To handle date and time operations.
* smtplib: To send emails using SMTP.
* email.mime.text: To format email content.
* configparser: To read configuration files.

**3. Configuration File**

* harvester.config: Contains paths, SMTP server details, and other configurations.

[Paths]

excel\_file\_path = path\_to\_excel\_file.xlsx

[SMTP]

server = smtp.example.com

port = 587

username = your\_email@example.com

password = your\_password

**4. Functions**

**4.1 send\_email**

* Purpose: Send an email to a recipient with optional CC recipients.
* Parameters:
  + recipient: Email address of the primary recipient.
  + cc\_recipients: List of CC recipients.
  + subject: Subject of the email.
  + body: Body of the email in HTML format.
  + smtp\_server: SMTP server address.
  + smtp\_port: SMTP server port.
  + smtp\_username: SMTP username.
  + smtp\_password: SMTP password.
* Process:
  + Create an email message using MIMEText.
  + Add subject, from, to, and cc fields.
  + Connect to the SMTP server and send the email.

**4.2 get\_latest\_date\_from\_path**

* Purpose: Retrieve the latest date from filenames in a given directory.
* Parameters:
  + path: Directory path containing the files.
* Process:
  + Check if the directory exists.
  + List all files in the directory.
  + Parse filenames to extract dates.
  + Return the latest date found.

**5. Main Workflow**

**5.1 Read Configuration File**

* Load Configurations: Read the harvester.config file to get paths and SMTP server details.

**5.2 Read Excel File**

* Load Data: Read the Excel file specified in the configuration.
* Cast Columns: Explicitly cast 'Last\_Delivery\_Date' to datetime.

**5.3 Update Last Delivery Dates**

* Iterate Rows: For each row, check the 'Data Delivery Location'.
* Get Latest Date: Use get\_latest\_date\_from\_path to find the latest date.
* Update DataFrame: Update 'Last\_Delivery\_Date' column with the latest date.

**5.4 Calculate Next Delivery Dates**

* Today's Date: Get the current date.
* Frequency Mapping: Map delivery frequencies to corresponding days.
* Iterate Rows: For each row:
  1. Calculate the next delivery date based on the frequency.
  2. Update 'ExpectedNextDate' column.
  3. Check if the next delivery date is overdue and set 'IsNeedEmailAlert' accordingly.

**5.5 Save Updated Data**

* Save Data: Save the updated DataFrame back to the Excel file.

**5.6 Send Email Alerts**

* Iterate Rows: For each row where 'IsNeedEmailAlert' is set to 1:
  1. Prepare the email content.
  2. Send emails to the recipient and CC recipients as specified.

**6. Error Handling**

* File Operations: Check if directories and files exist before processing.
* Email Sending: Catch and log any exceptions during the email sending process.

**7. Logging and Debugging**

* Print Statements: Include print statements to log key actions and errors for debugging purposes.

**8. Assumptions**

* The directory contains files named with dates in the format %m-%d-%y.
* The Excel file contains specific columns like 'Last\_Delivery\_Date', 'Data Delivery Location', 'NES\_Actual\_Frequency', 'ExpectedNextDate', 'IsNeedEmailAlert', 'Prodcode(s)', 'Publisher Name', 'CC\_mails', 'Reciver\_mails', and 'RepeatedCount'.
* The configuration file is correctly formatted and located in the same directory as the script.

**9. Example Configuration File**

[Paths]

excel\_file\_path = /path/to/your/excel\_file.xlsx

[SMTP]

server = smtp.yourserver.com

port = 587

username = your\_email@yourserver.com

password = your\_password