Project Name: PDF Parser Automation

Description: Automated pipeline for downloading, parsing, and staging PDF files with configurable workflows.

Author: Eva Iablocova

Repository: <https://github.com/EvaIablocova/PDF_Parser>

Date: 21.07.2025

Contents

[Project Overview 3](#_Toc204076547)

[Architecture 4](#_Toc204076548)

[Installation 6](#_Toc204076549)

[Usage 7](#_Toc204076550)

[Project location 7](#_Toc204076551)

[PDF\_parser\_sqls – folder with SQL scripts 7](#_Toc204076552)

[How to run the project 8](#_Toc204076553)

[Configuration 9](#_Toc204076554)

[There are 2 configuration files: 9](#_Toc204076555)

[config.json 9](#_Toc204076556)

[config\_last\_dates\_in\_db.json 10](#_Toc204076557)

[Scenarios for Using Configuration Files 11](#_Toc204076558)

[Slack 13](#_Toc204076559)

[Slack Notification Triggers: Three Scenarios for Sending Messages 13](#_Toc204076560)

[Adding new Slack recipient 13](#_Toc204076561)

# Project Overview

This project automates the process of downloading, parsing, and staging PDF files based on date changes and configuration settings. It identifies which files need to be processed, downloads them, parses their contents, loads the results into a database, and manages file archiving. The workflow is controlled by command-line arguments and configuration files, with logging at each step for traceability. The main technologies used are Python and SQL.

# Architecture

A diagram of a flowchart

AI-generated content may be incorrect.

**Pic. 1 -** **Data Flow**

The project is structured as a modular pipeline for processing PDF files:

Dates are loaded and compared.

Changed PDFs are downloaded.

Files are parsed, cleaned, and loaded into the database.

Config is updated and files are archived.

Staging and post-processing are performed (SQL job – merge and validate data, send Slack message)

# Installation

1. Clone the repository to your local machine:

git clone <repository\_url>

1. Install Python 3.8 or newer.
2. Install required dependencies:

pip install -r requirements

1. Configure the project:

Edit config.json to set paths for download\_dir, parsed\_data\_dir, and other settings.

Ensure all required scripts (1\_load\_dates\_from\_site.py, 3\_1\_parser.py, etc.) are present in the project directory.

1. Run the entry point script:

python 0\_1\_entry\_point.py [download| parse | stage]

Use **download** to start from downloading files.

Use **parse** to start from parsing downloaded files.

Use **stage** to stage parsed data.

1. Check logs for progress and errors.

# Usage

## Project location

A screenshot of a computer

AI-generated content may be incorrect.

**Pic.2 - Project location**

## PDF\_parser\_sqls – folder with SQL scripts

A screenshot of a computer program

AI-generated content may be incorrect.

**Pic.3 - PDF\_parser\_sqls**

**0\_Deployment\_script.sql** –one run time script for creating the structure and constraints

**1\_Merge.sql** – the first step in SQL job

**2\_Validation.sql** – the second step in SQL job

**3\_agregate\_message.sql** – the third step in SQL job

**4\_call\_sp\_to\_check\_logs.sql** - the fourth step in SQL job

## How to run the project

**1 variant:**

1. Choose the operation mode: download, parse, or stage (by default – download)

2. Run the entry point script with the desired mode:

python 0\_1\_entry\_point.py [download|parse|stage]

A black screen with white text

AI-generated content may be incorrect.

**download**: Downloads new or changed PDF files based on date comparison.

**parse**: Parses downloaded files (all files in “files\_to\_parse” folder), cleans data, loads results into the database, and archives processed files.

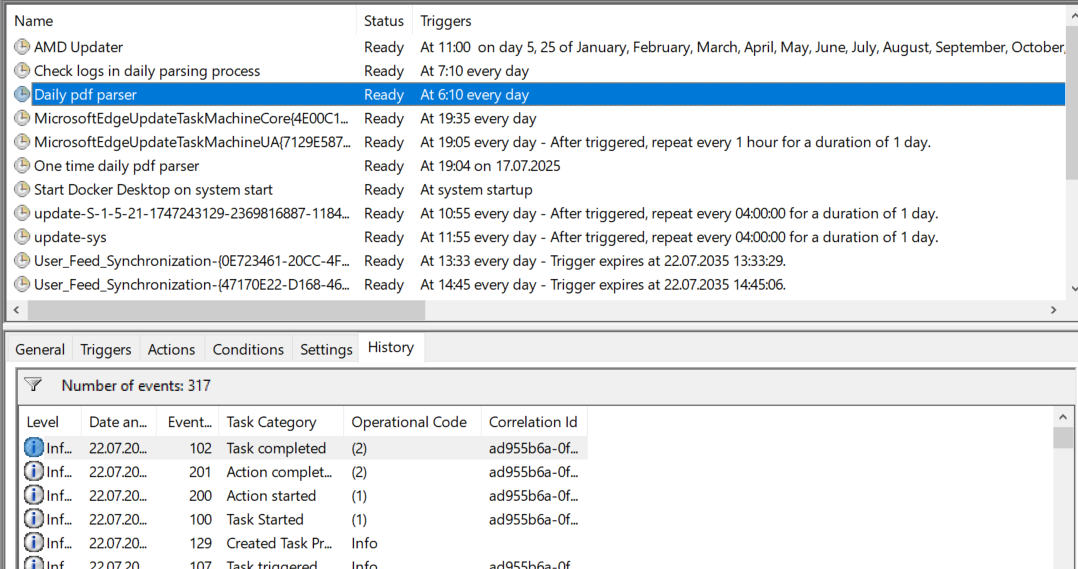
**stage**: Stages parsed data (all files in “parsed\_files” folder) for further processing and updates configuration.

3. Monitor progress and errors in the log messages generated during each step.

4. Check the output directories (download\_dir, parsed\_data\_dir) for processed files.

**2 variant:**

1. Go to the scheduler
2. Find “Daily pdf parser” job and run it

****

**Pic. 4 – Windows job**

# Configuration

## There are 2 configuration files:

A screenshot of a computer

AI-generated content may be incorrect.

**Pic.5 – Configuration files**

## config.json

**config.json** – contains all settings parameters for parsing files

Edit the config.json file to set the following parameters:

**download\_dir:** Directory for downloaded PDF files.

**parsed\_data\_dir:** Directory for parsed data files.

**config\_last\_dates\_in\_db**: Stores the last processed dates for each file.

**today\_file:** Path to the file containing today's dates.

**file\_configs:** List of dictionaries specifying keywords and parsing settings for each file type.

A screenshot of a computer program

AI-generated content may be incorrect.

**Pic. 6 – part of config.json file**

## config\_last\_dates\_in\_db.json

config\_last\_dates\_in\_db.json – contains last updated dates for each file.

A screenshot of a computer program

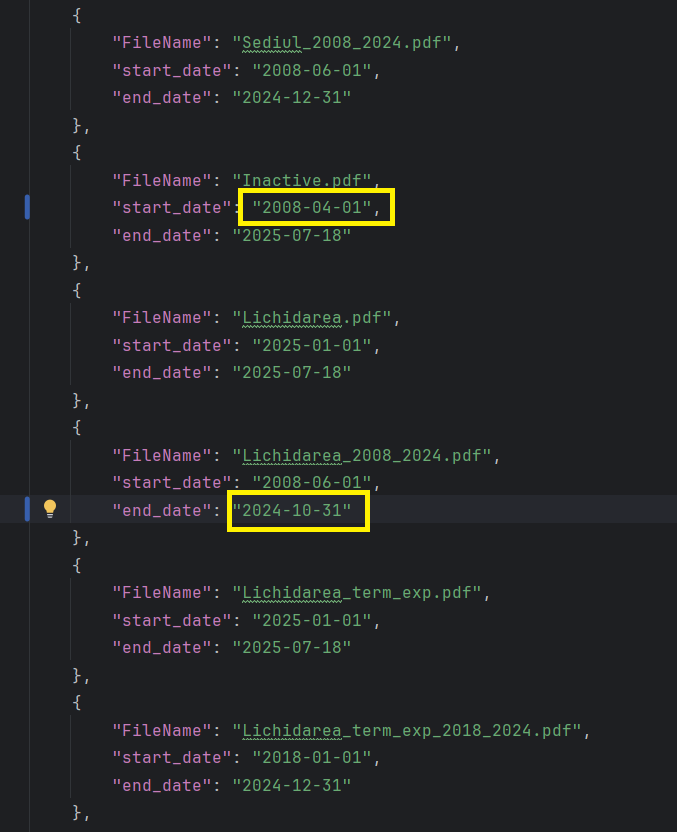
AI-generated content may be incorrect.

**Pic. 7 – part of config\_last\_dates\_in\_db.json file**

## Scenarios for Using Configuration Files

* If we need to download 1 or several files, we have to change ether “start\_time” or “end\_time” for these files and run the project (or Windows job “Daily pdf parser”).

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Pic. 8 – Before and after changings for a few files**

* If we need to make full run, we just can insert into “config\_last\_dates\_in\_db.json” instead of all information “{}”.

A screenshot of a computer program

AI-generated content may be incorrect.A screen shot of a computer

AI-generated content may be incorrect.

**Pic. 9** **– Before and after changings for all files**

# 

# Slack

## Slack Notification Triggers: Three Scenarios for Sending Messages

1. When task completed successfully and files were downloaded

A screenshot of a computer

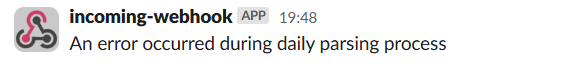
AI-generated content may be incorrect.

1. When Task completed successfully and files were not downloaded, because dates were not changed on the site

A black text on a white background

AI-generated content may be incorrect.

1. When Task completed with error



## Adding new Slack recipient

We have table for Slack configuration settings

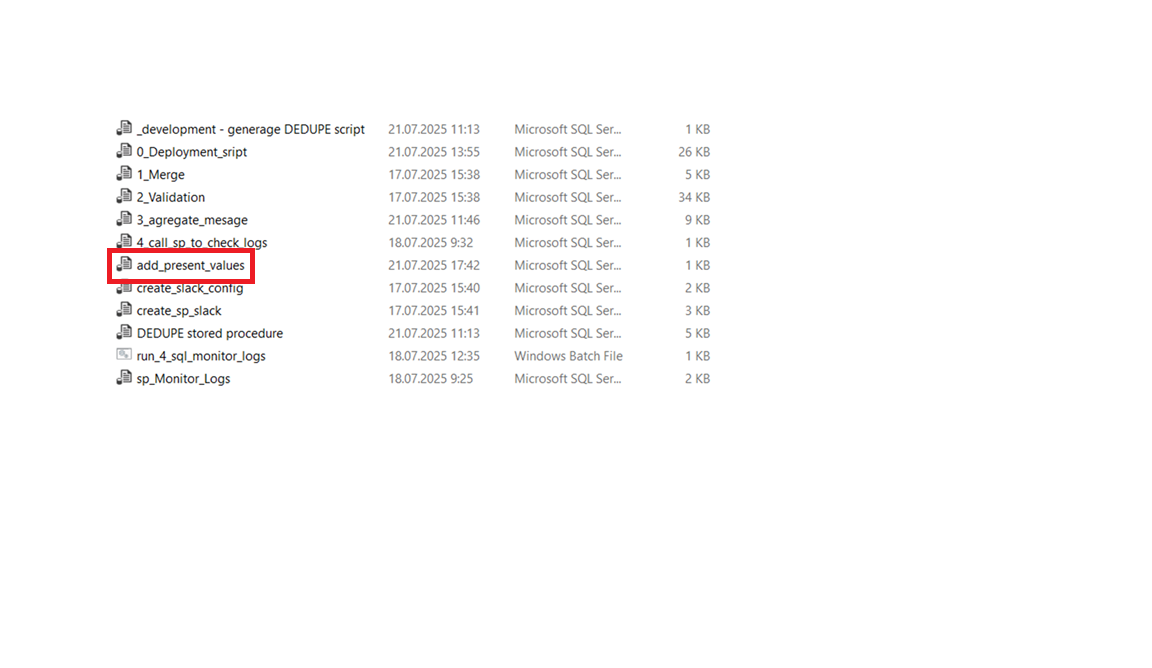
A screenshot of a chat

AI-generated content may be incorrect.

**Pic. 10 – SlackConfigSetting\_SID table**

If we need to add new Slack recipient:

1. We need to execute this procedure with valid parameters

A screenshot of a computer program

AI-generated content may be incorrect.

**Pic. 11 – add\_present\_values for Slack messages recipient**

1. And change the “Recipient” in SQL job “validate\_and\_insert\_data”

**A screenshot of a computer

AI-generated content may be incorrect.**

**Pic. 12 – Open “validate\_and\_insert\_data” SQL job**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Pic. 13 – Open “send\_agregate\_message” step**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Pic. 14 – Change the “Recipient’ at the end of script**