Context

This is to test your proficiency as a software developer

Challenge

Create a script to parse PDF files of court cases to extract the data required to form a metadata json file for the case (sample of metadata file at this link).

In addition, two other files are to be generated that hold the content of the case files as show in their samples

- A technical txt file (sample)
- A preview md file (sample)

NB: *the technical txt file makes use of certain delimiters for headings (__section__) and paragraphs (__paragraph__) so follow suit as shown in the sample file*

Please create the script with Javascript/Typescript or Python. You are free to (and encouraged) to use any other tools, libraries or packages.

Here is the required user story:

1. A given number of PDF files of court cases can be read from the filesystem by the script and parsed, generating a set of three files for of each case: (1) preview.md, (2) technical.txt and (3) metadata.json

2. An error log file is outputted with information about
   ○ Any file(s) that failed to process
   ○ The specific error that occurred

3. An info log file is outputted with information about metadata files that have missing, empty or erroneous fields after processing, with information about
   ○ The name of the source file
   ○ The name of the generated metadata file
   ○ A list of missing/empty/erroneous fields alongside an error message describing the issue encountered

Technical requirements:

- The input PDF files can be one of two types in these two folders ([samples here](#)). Also, note that some PDFs are scanned PDFs. You may decide to build different parsers to handle the two types uniquely.
- Each case has a unique suit no/civil appeal no/criminal appeal no as the *Case Number*. The output files should be renamed accordingly.

NB: Aside log files, all other generated files should be named with the following format

<Short Title>_<Case Number>_<TimeStamp>_<Generated FIle Type>.<FIle extension>

Submission

Make your submission via email. For your submission, include the following:

- A link to a public Git repository with your project code
- A 1 page technical description of your submission (Template here)

Ideally, include all the Git commits you have made throughout the development. (i.e., commit as you're developing instead of pushing everything at the end).

Evaluation Criteria (in the order of importance):

- Fulfilling the requirements
- Structure of the code
- Readability of the code
- Using best practices
- Additional points for building one parser which can parse input files of either type.