Assignment Task: PDF Parsing and Structured JSON Extraction

Objective

Your task is to build a **Python program** that parses a PDF file and extracts its content into a **well-structured JSON format**. The extracted JSON must preserve the **hierarchical organization** of the document (e.g., sections, sub-sections, and content blocks) while clearly identifying different data types.

Requirements

1. **Input & Output**
    - Input: A PDF file (sample provided below).
    - Output: A JSON file containing the extracted content.
2. **JSON Structure**

    The JSON must:

    - Maintain **page-level hierarchy**.
    - Capture the **type of data**:
        - paragraph
        - table
        - chart
    - Include **section and sub-section names** where applicable.
    - Ensure that text is extracted in a **clean and readable format**.

Example JSON (illustrative only):

```
{
  "pages": [
    {
      "page_number": 1,
      "content": [
        {
          "type": "paragraph",
          "section": "Introduction",
          "sub_section": "Background",
          "text": "This is an example paragraph extracted from the PDF..."
        },
        {
          "type": "table",
          "section": "Financial Data",
          "description": null,
          "table_data": [
            ["Year", "Revenue", "Profit"],
            ["2022", "$10M", "$2M"],
            ["2023", "$12M", "$3M"]
          ]
        },
        {
          "type": "chart",
          "section": "Performance Overview",
          "Table_data": [
             [XLabel, YLabel],
             ["2022", "$10M"],
            ["2023", "$12M",]
          "description": "Bar chart showing yearly growth..."
        }
      ]
    }
  ]
}
```

3. **Implementation Guidelines**

    - You may use **any Python libraries/tools** for parsing and extraction (e.g., pdfplumber, PyMuPDF, camelot, pytesseract, pdfminer, etc.).

    - Your program should be modular, cleanly structured, and well-documented.

    - The solution must be **robust** enough to handle different types of content.

4. **Deliverables**

    - A Python script (.py file) that takes a PDF file as input and produces a JSON file as output.

    - A brief README with instructions on:

        - How to install dependencies.

        - How to run the program.

Sample PDF File

Download PDF

Evaluation Criteria

- **Accuracy** of extracted content.
- **Correctness** of JSON structure and hierarchy.