INTRODUCTION ABOUT THE INDUSTRY

About the company

Coal India Limited (**CIL**) is an Indian state-controlled coal mining company headquartered in Kolkata, West Bengal, India and the largest coal-producing company in the world and a Maharatna company.

The company contributes to around 82% of the coal production in India. It produced 554.14 million tonnes of raw coal in 2016-17, an increase from its earlier production of 494.24 million tonnes of coal during FY2014–15 and earned a revenue of Rs. 95,435 crore(US$14 billion) from sale of coal in the same financial year. As on 14 October 2015, Union Government of India owns CIL and controls the operations of CIL through Ministry of Coal. In April 2011, CIL was conferred the Maharatna status by the Union Government of India, making it one of the seven maharatnas. As on 14 October 2015, its market capitalization was Rs. 2.11 lakh crore (US$31 billion) making it India's 8th most valuable company by market value.

Corporate structure and subsidiary companies

Coal India is a holding company with seven wholly owned coal producing subsidiary companies and one mine planning and consultancy company. It encompasses the whole gamut of identification of coal reserves, detailed exploration followed by design and implementation and optimizing operations for coal extraction in its mines. The producing companies are:

- Eastern Coalfields Limited (ECL), Sanctoria, West Bengal
- Bharat Coking Coal Limited (BCCL), Dhanbad, Jharkhand
- Central Coalfields Limited (CCL), Ranchi, Jharkhand
- South Eastern Coalfields Limited (SECL), Bilaspur, Chattisgarh
- Western Coalfields Limited (WCL), Nagpur, Maharashtra
- Northern Coalfields Limited (NCL), Singrauli, Madhya Pradesh
- Mahanadi Coalfields Limtied (MCL), Sambalpur, Orissa

The consultancy company is Central Mine Planning and Design Institute Limited (CMPDIL), Ranchi, Jharkhand.

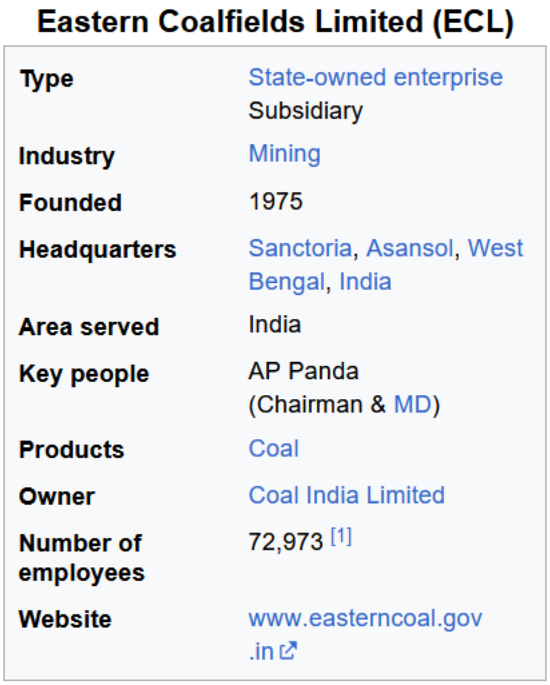
GEOGRAPHY:

ECL mining leasehold area is 753.75 km^2 and surface right area is 237.18 km$^{2.}$It is spread across West Bengal and Jharkhand. ECL operates in Raniganj Coalfield in West Bengal, Mugma field (in Dhanbad district), Rajmahal Coalmines Projects (in Godda district) and Chitra-Saharjuri Coalfield (in Deoghar district) in Jharkhand. The heart of the Raniganj Coalfield is located south of the Ajay River and north of the Damodar River in Paschim Bardhaman district. Mejia (in Bankura district) and Parbelia (in Purulia district) are located south of the Damodar. Palasthali (in Birbhum district) is located north of the Ajay. The Mugma field is located west of the Barakar River and north of the Damodar.The formation of coal seams in the Raniganj Coalfield-Mugma field of ECL has occurred mainly in Raniganj measures and Barakar measures. The Santhal Pargana mines and the Rajmahal Area are mainly related to Barakar measures and Talchair series

.

Reserves

As of 1 April 2012, ECL has proved reserves of 16.94 billion tonnes – 12.42 billion tonnes in West Bengal and 4.52 billion tonnes in Jharkhand. Total reserves in the ECL command area, up to a depth of 600 m, was 49.17 billion tonnes.



Eastern Coalfields Limited (ECL)

Type	State-owned enterprise Subsidiary
Industry	Mining
Founded	1975
Headquarters	Sanctoria, Asansol, West Bengal, India
Area served	India
Key people	AP Panda (Chairman & MD)
Products	Coal
Owner	Coal India Limited
Number of employees	72,973 [1]
Website	www.easterncoal.gov.in

PYTHON PROGRAM TO CONVERT PDF TO .txt FORMAT

The Python script is a simple PDF to text converter using the PyPDF2 library.

1. Class Definition:

- Defines a class named `PdfToTxtConverter`.

- Initializes the class with the PDF file path (`pdf_path`) and the desired output file path (`output`).

2. Conversion Method:

- Defines a method named `convert_to_txt` within the class.

- Opens the PDF file in binary mode (`'rb'`) using PyPDF2.

- Checks if the PDF has pages. If not, it prints a message and returns.

- Iterates through each page of the PDF, extracts text, and concatenates it into the `text` variable.

- Writes the extracted text to the specified output text file (`self.output`).

- Prints a message indicating that the text data has been written to the output file.

3. Main Block:

- In the main block, it specifies the input PDF file path (`pdf_path`) and the output text file path (`output_path`).

- Creates an instance of the `PdfToTxtConverter` class with the specified paths.

- Calls the `convert_to_txt` method to perform the PDF to text conversion.

So, when we run this script, it reads the specified PDF file, extracts text from each page, and writes the concatenated text to the output text file. The `print` statements provide information about the process, including any errors that may occur.

PYTHON PROGRAM TO CONVERT AN OCR PDF TO .txt FORMAT

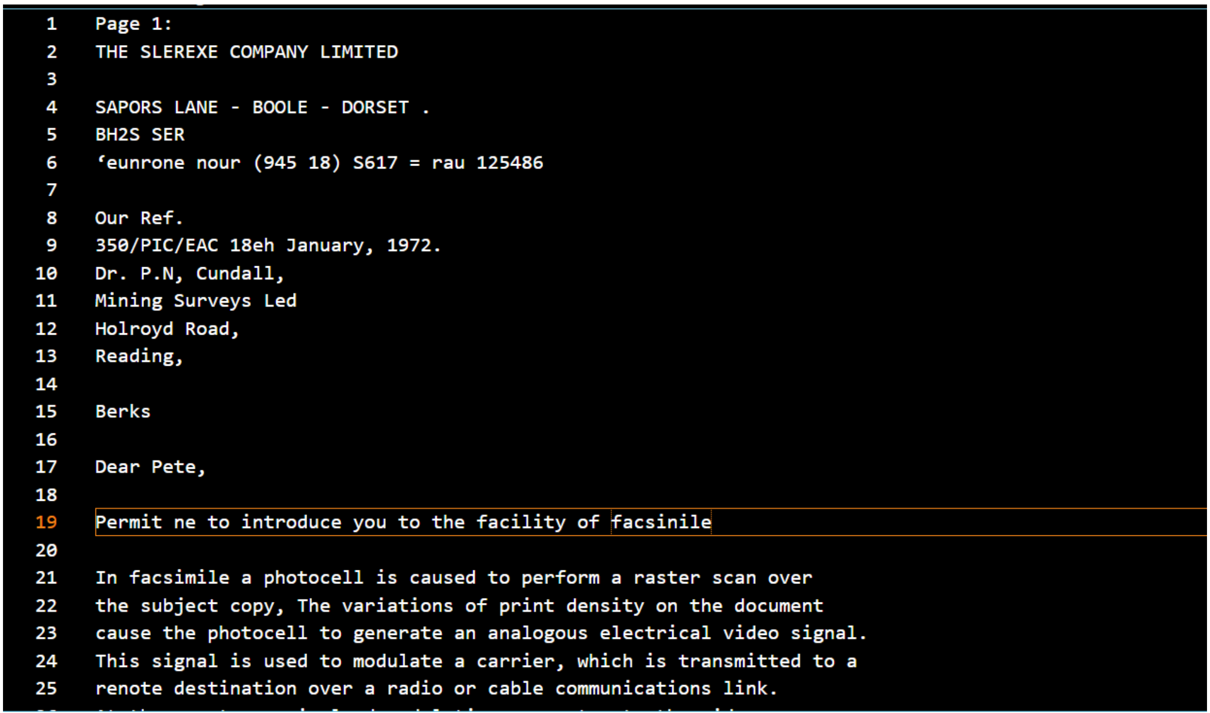
METHODOLOGY:

1. The `PdfOcrExtractor` class is defined to encapsulate the PDF extraction functionality.
2. The `extract_text` method opens the PDF file, iterates through its pages, converts each page to an image using PyMuPDF, and then uses pytesseract to perform OCR on the image to extract text. The extracted text is then concatenated and returned.
3. The `save_text_to_file` method takes the extracted text and saves it to a specified text file.
4. In the `__main__` block, a PDF file path (`pdf_file_path`) and an output text file path (`output_txt_path`) are specified.
5. An instance of the `PdfOcrExtractor` class is created with the specified paths.
6. The `extract_text` method is called to extract text from the PDF.
7. If text extraction is successful, the extracted text is printed and saved to a text file using the `save_text_to_file` method. If not, an error message is printed.

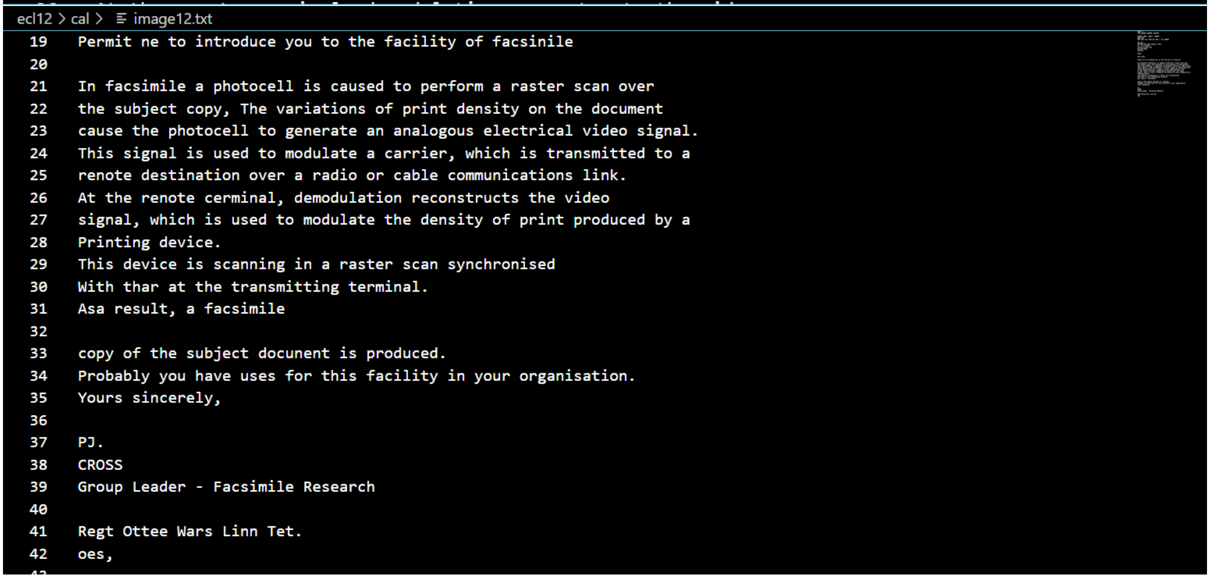
REQUIREMENTS FOR THE PROGRAM:

- Make sure we have the required libraries installed (`fitz`, `pytesseract`, `PIL`, and `nltk`).
- We can install them using `pip install PyMuPDF pytesseract Pillow nltk`.
- Ensure that Tesseract OCR is installed on our system. The script relies on Tesseract, and it needs to be available in our PATH.
- Depending on the complexity and quality of the PDF, the OCR results may vary. Adjustments to OCR settings or pre-processing of images may be necessary for better results.
- Consider handling potential exceptions more gracefully and providing more informative error messages to aid in troubleshooting.
- Ensure the PDF file (`Scansmpl.pdf` in our case) is present in the specified path.
- Keep in mind that OCR might not always produce perfect results, especially with complex layouts or poor image quality. Fine-tuning and experimenting with OCR settings may improve accuracy.

SAMPLE OUTPUT:



```
Page 1:
THE SLEREXE COMPANY LIMITED

SAPORS LANE - BOOLE - DORSET .
BH2S SER
‘eunrone nour (945 18) S617 = rau 125486

Our Ref.
350/PIC/EAC 18eh January, 1972.
Dr. P.N, Cundall,
Mining Surveys Led
Holroyd Road,
Reading,

Berks

Dear Pete,

Permit ne to introduce you to the facility of facsinile

In facsimile a photocell is caused to perform a raster scan over
the subject copy, The variations of print density on the document
cause the photocell to generate an analogous electrical video signal.
This signal is used to modulate a carrier, which is transmitted to a
renote destination over a radio or cable communications link.
```



```
Permit ne to introduce you to the facility of facsinile

In facsimile a photocell is caused to perform a raster scan over
the subject copy, The variations of print density on the document
cause the photocell to generate an analogous electrical video signal.
This signal is used to modulate a carrier, which is transmitted to a
renote destination over a radio or cable communications link.
At the renote cerminal, demodulation reconstructs the video
signal, which is used to modulate the density of print produced by a
Printing device.
This device is scanning in a raster scan synchronised
With thar at the transmitting terminal.
Asa result, a facsimile

copy of the subject docunent is produced.
Probably you have uses for this facility in your organisation.
Yours sincerely,

PJ.
CROSS
Group Leader - Facsimile Research

Regt Ottee Wars Linn Tet.
oes,
```

DJANGO:

Django is a high-level web framework written in Python that encourages rapid development and clean, pragmatic design. It follows the Model-View-Controller (MVC) architectural pattern, but in Django, this is referred to as the Model-View-TemplateMVT) pattern. Django provides a set of tools and conventions for building web applications, making it easier to develop, deploy, and maintain complex web applications.
Here are some key features and concepts in Django:

1. **Object-Relational Mapping (ORM)**: Django provides an ORM system that allows you to define your database schema using Python classes. It abstracts the database layer, making it easier to interact with databases without writing raw SQL queries.

2. Admin Interface: Django comes with a built-in admin interface that can be easily customized. This interface allows you to manage your application's data models, making it simple to perform CRUD (Create, Read, Update, Delete) operations.

3. **URL Routing and Views:** Django uses a URL routing system to map URLs to views. Views are Python functions or classes that handle HTTP requests and return HTTP responses. URL patterns are defined in a central `urls.py` file.

4. **Templates:** Django uses a template system to separate the HTML code from the Python code. Templates are used to generate HTML dynamically and can include variables, loops, and conditional statements.

5. **Middleware**: Django includes a middleware system that allows you to process requests and responses globally.

6. **Forms:** Django provides a form-handling system that simplifies the process of handling HTML forms, including validation and rendering.

7. **Authentication and Authorization**: Django includes built-in tools for user authentication and authorization. It supports user registration, login, and password recovery.

8**. Security:** Django includes security features such as protection against common web attacks like Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF).

9. **RESTful API Support**: While Django is primarily designed for building traditional web applications, it also provides tools for building RESTful APIs.

10. **Middleware**: Django includes middleware components that allow you to process requests and responses globally. This can be used for tasks such as authentication, security, and caching.

To get started with Django, we can install it using pip:

Bash:
pip install Django

After installation, we can create a new Django project and start building our web application. The official Django documentation (https://docs.djangoproject.com/) is an excellent resource for learning and mastering Django.

Image to text converter – what is this?

OCR tool is the Image to text converter based on Optical character recognition technology.

OCR tools use cases:

Searchable PDF
The most useful feature is converting a scanned PDF into a searchable PDF. This option allows you to quickly find the necessary information in the extracted text. This function is often used by libraries and government agencies to digitize their archives.

Education
Teachers and students can convert scanned study notes, textbooks and lecture notes into text for better exam preparation. Scanned lectures takes a lot of space on your hard drive or phone. The text-based version, created via image to text converter takes up much less space.

Book Digitization
Book digitization is the process of converting physical books, magazines and other records into digital media using an image to text converter. As content digitizes, more and more publishers and organizations are digitizing their physical books into text formats such as PDF/A for easy distribution and reproduction in the online space. These digitized books can then be read on a digital screen. The editable format helps reduce file size and allows third-party applications to search, reformat, or manipulate text.

Data mining
Data mining is the process of extracting and discovering patterns in large data sets using methods that intersect machine learning, statistics, and database systems. Image to text conversion is the first step in preparing structured information to data mining set.

Data extraction
It is used to extract text from invoices, receipts, tables, forms and other documents to create databases and spreadsheets.

Quick translation
When you see something in an unknown language, you can simply take a photo, send it to our image to text conversion service, and once processed, send the extracted data to the translator.

Legislation and Compliance
Typically, legal documents are got in scanned form. Using picture to text converter you can extract important information from legal documents, contracts, invoices or government docs. Image to text converter gives you the ability to convert scanned documents into digital versions.

METHODOLOGY:

The methodology used in an OCR to text (TXT) API involves several key steps.

1. **Image Input**:
   - Users provide an image containing text as input to the OCR API.

2. **Image Preprocessing:**
   - The input image may undergo preprocessing steps to enhance its quality. This can include operations like resizing, normalization, noise reduction, and contrast adjustment.

3.**Text Detection**:
   - The API identifies the regions within the image that contain text. This step involves detecting bounding boxes or polygons around text elements.

4. **Text Recognition**:
   - The text within the detected regions is recognized using Optical Character Recognition (OCR) algorithms. These algorithms analyze the shapes and patterns of the characters and convert them into machine-readable text.

5. **Language Understanding:**
   - Some OCR systems may incorporate language understanding capabilities to improve accuracy. This involves analyzing the context of the recognized text to better interpret ambiguous characters or words.

6. **Post-Processing**:
   - The recognized text may undergo post-processing to correct errors, improve formatting, and handle special cases like tables or multi-column layouts.

7. **Output Formatting**:
   - The final step involves formatting the recognized text into a user-friendly format, such as plain text (TXT), and making it available as the API output.

8. **API Response**:
   - The API returns the processed and formatted text to the user, making it accessible for further use or integration into other applications.

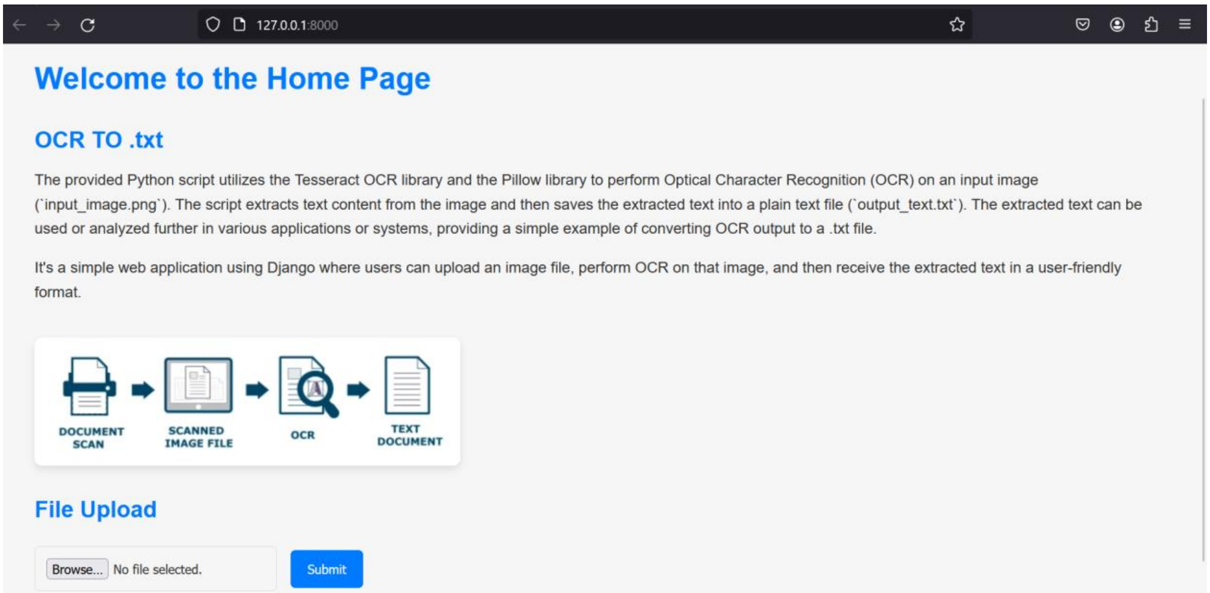
9. **Error Handling:**
   - The API should include mechanisms for handling errors gracefully, providing informative error messages, and ensuring a robust user experience.

10. **Security Measures:**
   - Depending on the implementation, the OCR to text API should also incorporate security measures to protect user data and ensure the confidentiality of the processed information.

It's important to note that different OCR systems may use variations of these steps, and the effectiveness of the OCR process can depend on factors such as the quality of the input image, the language of the text, and the specific algorithms employed by the OCR engine.

RESULT:

HOME PAGE :



Welcome to the Home Page

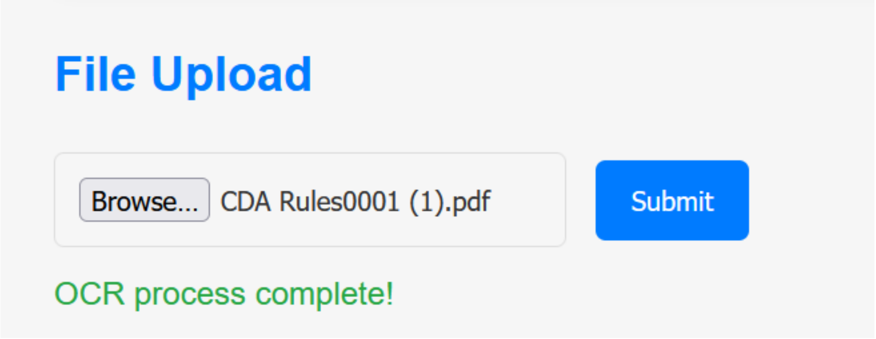
OCR TO .txt

The provided Python script utilizes the Tesseract OCR library and the Pillow library to perform Optical Character Recognition (OCR) on an input image (`input_image.png`). The script extracts text content from the image and then saves the extracted text into a plain text file (`output_text.txt`). The extracted text can be used or analyzed further in various applications or systems, providing a simple example of converting OCR output to a .txt file.

It's a simple web application using Django where users can upload an image file, perform OCR on that image, and then receive the extracted text in a user-friendly format.

File Upload

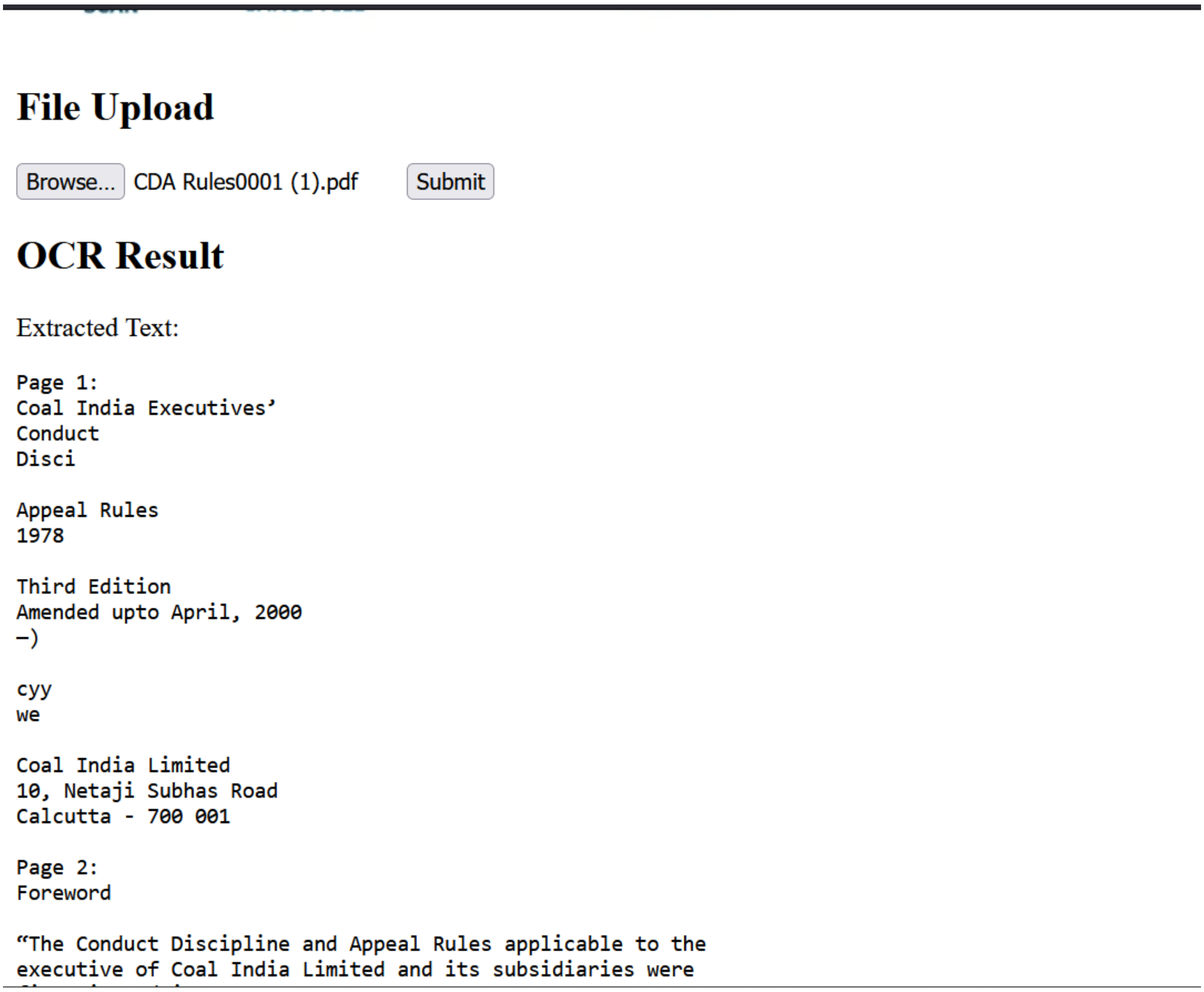
OUTPUT PAGE:



File Upload

CDA Rules0001 (1).pdf

OCR process complete!



```
Page 1:
Coal India Executives’
Conduct
Disci

Appeal Rules
1978

Third Edition
Amended upto April, 2000
—)

cyy
we

Coal India Limited
10, Netaji Subhas Road
Calcutta - 700 001

Page 2:
Foreword

“The Conduct Discipline and Appeal Rules applicable to the
executive of Coal India Limited and its subsidiaries were
```

Future Enhancements:

1. **Language Support:**
   - **Enhancement**: Expand language support for OCR to cover a broader range of languages and dialects.
   - **Rationale**:Cater to a more diverse user base with documents in different languages.

2.**Advanced Image Preprocessing**:
   - **Enhancement**:Implement advanced image preprocessing techniques for better handling of low-quality or distorted images.
   - **Rationale**:Improve OCR accuracy and performance in challenging image conditions.

3. **Deep Learning Integration**
   - **Enhancement**:Investigate and integrate state-of-the-art deep learning models for OCR to enhance accuracy.
   - **Rationale:** Leverage advancements in machine learning for improved text recognition.

4. **Cloud Integration**:
   - **Enhancement**: Develop cloud-based OCR services with seamless integration with popular cloud platforms.
   - **Rationale**:Facilitate scalability, accessibility, and ease of deployment for users.

5. **Interactive OCR Training**:
   - **Enhancement:**Implement interactive OCR model training features, allowing users to fine-tune models based on specific domains or document types.
   - **Rationale**: Provide users with the ability to customize OCR models for their unique requirements.

6. **Natural Language Processing (NLP) Integration**:
   -**Enhancement:** Integrate NLP techniques for post-OCR analysis, including entity recognition and sentiment analysis.
   - **Rationale**: Enhance the value of extracted text by providing deeper insights into its meaning.

7. **Optimized Resource Utilization:**
   - **Enhancement**: Optimize resource utilization by implementing resource-aware algorithms and efficient memory management.
   - **Rationale**:Improve overall performance and reduce infrastructure costs for users.

8. **Parallel Processing Improvements**:
   - **Enhancement:** Fine-tune parallel processing capabilities to further enhance API speed and efficiency.
   - **Rationale:**Continue to explore opportunities for performance optimization in handling multiple image requests.

9. **User Feedback Mechanism:**
   - **Enhancement**: Implement a user feedback mechanism to collect user experiences and improve OCR accuracy based on real-world usage.
   - **Rationale:**Continuously refine and enhance the OCR models based on user feedback and evolving requirements.

10. **Integration with Emerging OCR Standards**:
    - **Enhancement**: Stay abreast of emerging OCR standards and ensure compatibility and compliance with industry best practices.
    - **Rationale**: Position OCRPyText API as a cutting-edge solution aligned with evolving industry standards.

CONCLUSION:

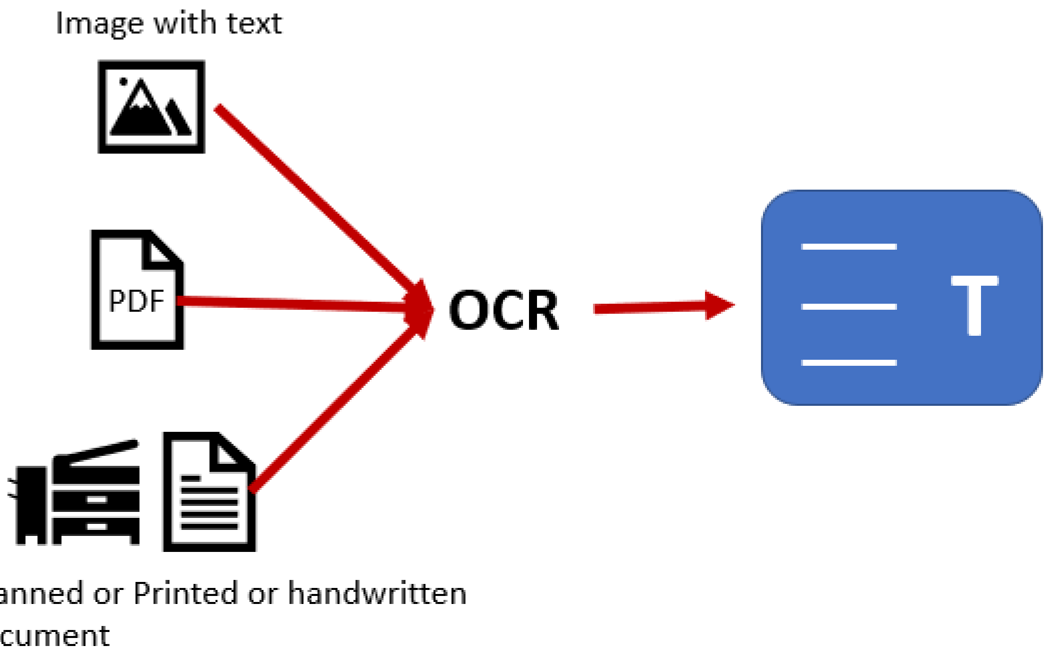
OCR API is a cloud-based service that provides web interfaces to integrate Optical Character Recognition (OCR) technology into your software application or web site. OCR Web Service is efficient, powerful and scalable platform capable of processing huge volumes of images and documents.

OCR Web Service allows you to:

Convert image to text or extract text and characters from scanned PDFs or images. Service includes Zonal OCR to extract specifically zoned text from image.

Convert OCRed results directly to editable formats like txt.

Get extracted text or converted file and send to Database directly or our executable program.



REFERENCES:

https://en.wikipedia.org/wiki/Eastern_Coalfields

https://geekflare.com/convert-image-to-text/

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwiVpauhmdeDAxWnkq8BHV1CDfoQFnoECC8QAQ&url=https%3A%2F%2Fnanonets.com%2Fblog%2Fconvert-image-to-text%2F&usg=AOvVaw0NxVrxwIE4c9Miq2wCLepA&opi=89978449

https://pypi.org/project/pytesseract/

https://nanonets.com/blog/ocr-with-tesseract/

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwi9w-fBmdeDAxUJcPUHHf49AgMQFnoECCwQAQ&url=https%3A%2F%2Fwww.geeksforgeeks.org%2Ftext-localization-detection-and-recognition-using-pytesseract%2F&usg=AOvVaw2fAQmj1toqVYSbGfUgSISx&opi=89978449

https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwi9w-fBmdeDAxUJcPUHHf49AgMQFnoECCoQAQ&url=https%3A%2F%2Fwww.analyticsvidhya.com%2Fblog%2F2021%2F12%2Foptical-character-recognition-using-pytesseract%2F&usg=AOvVaw25x9FPEmWaVREkscE_Jv7j&opi=89978449

THANK YOU