

VIRTUNEXA PYTHON DEVELOPMENT TASK-1

PROGRAM:

Install necessary packages if not already installed

!pip install PyPDF2 python-docx

import re

import os

import PyPDF2

from docx import Document

from google.colab import files

----- Resume Extraction Functions -----

def extract_text(file_path):

if not os.path.exists(file_path):

raise FileNotFoundError("File not found.")

if file_path.lower().endswith(".pdf"):

with open(file_path, 'rb') as file:

reader = PyPDF2.PdfReader(file)

return ' '.join(page.extract_text() for page in reader.pages if
page.extract_text())

elif file_path.lower().endswith(".docx"):

doc = Document(file_path)

return ' '.join(para.text for para in doc.paragraphs)

else:

raise ValueError("Unsupported file format. Use PDF or DOCX.")

def extract_info(text):

name = re.findall(r"Name[:\-\]?\s*([A-Z][a-z]+(?:\s+[A-Z][a-z]+)+)", text)

email = re.findall(r"[a-zA-Z0-9_\.+-]+\@[a-zA-Z0-9]+\.[a-zA-Z0-9-\.]+", text)

phone = re.findall(r"\+?\d[\d\s\-\]{8,15}", text)

skills = re.findall(r"Skills[:\-\]?\s*(.*)", text, re.IGNORECASE)

return {

"Name": name[0] if name else "Not found",

"Email": email[0] if email else "Not found",

"Phone": phone[0] if phone else "Not found",

"Skills": skills[0] if skills else "Not found"

}

----- Calculator Function with Logging -----

def calculate(a, b, operator):

try:

a = float(a)

b = float(b)

if operator == '+':

result = a + b

elif operator == '-':

```

        result = a - b
    elif operator == '*':
        result = a * b
    elif operator == '/':
        if b == 0:
            raise ZeroDivisionError("Cannot divide by zero.")
        result = a / b
    else:
        raise ValueError("Invalid operator.")

    # Log operation to file
    with open("calculation_history.txt", "a") as log_file:
        log_file.write(f"{a} {operator} {b} = {result}\n")

    return result

except Exception as e:
    return f"Error: {str(e)}"

# ----- Main Program -----

def main():
    print("=== Resume Info Extractor & Calculator ===")
    print("Choose an option:")
    print("1. Extract Resume Info (PDF/DOCX)")
    print("2. Use Calculator")

```


```
choice = input("Enter 1 or 2: ")

if choice == "1":
    print("Upload a PDF or DOCX file")
    uploaded = files.upload()
    file_path = next(iter(uploaded))

    try:
        text = extract_text(file_path)
        info = extract_info(text)
        print("\nExtracted Information:")
        for key, value in info.items():
            print(f"{key}: {value}")
    except Exception as e:
        print(f"Error: {e}")

elif choice == "2":
    a = input("Enter number A: ")
    b = input("Enter number B: ")
    operator = input("Enter operation (+, -, *, /): ")
    result = calculate(a, b, operator)
    print("Result:", result)

else:
    print("Invalid choice. Please enter 1 or 2.")
```

 Correct name guard for execution

```
if __name__ == "__main__":
```

```
    main()
```

```
Requirement already satisfied: PyPDF2 in /usr/local/lib/python3.11/dist-packages (3.0.1)
Requirement already satisfied: python-docx in /usr/local/lib/python3.11/dist-packages (1.1.2)
Requirement already satisfied: lxml>=3.1.0 in /usr/local/lib/python3.11/dist-packages (from python-docx) (5.4.0)
Requirement already satisfied: typing-extensions>=4.9.0 in /usr/local/lib/python3.11/dist-packages (from python-docx) (4.13.2)
=== Resume Info Extractor & Calculator ===
Choose an option:
1. Extract Resume Info (PDF/DOCX)
2. Use Calculator
Enter 1 or 2: 1
Upload a PDF or DOCX file
Choose Files resume.pdf
• resume.pdf(application/pdf) - 1139 bytes, last modified: 6/1/2025 - 100% done
Saving resume.pdf to resume.pdf

Extracted Information:
Name: Alice Smith
Email
Email: alice.smith@gmail.com
Phone: +91 9876543210

Skills: Python, Machine Learning, TensorFlow, SQL
```

```
Requirement already satisfied: PyPDF2 in /usr/local/lib/python3.11/dist-packages (3.0.1)
Requirement already satisfied: python-docx in /usr/local/lib/python3.11/dist-packages (1.1.2)
Requirement already satisfied: lxml>=3.1.0 in /usr/local/lib/python3.11/dist-packages (from python-docx) (5.4.0)
Requirement already satisfied: typing-extensions>=4.9.0 in /usr/local/lib/python3.11/dist-packages (from python-docx) (4.13.2)
=== Resume Info Extractor & Calculator ===
Choose an option:
1. Extract Resume Info (PDF/DOCX)
2. Use Calculator
Enter 1 or 2: 2
Enter number A: 2
Enter number B: 6
Enter operation (+, -, *, /): *
Result: 12.0
```