Code for Quick Data Analysis

```
import os
import pandas as pd
import openai
from docx import Document

def ask_api_key():
    """
    Prompt for the OpenAI API key.
    """
    return input("Enter your OpenAI API key: ")

def analyze_file(file_path, api_key):
    """
    Analyze the provided file and generate a dataset analysis.
    """
    # Load the dataset
    if file_path.endswith('.csv'):
        df = pd.read_csv(file_path)
    elif file_path.endswith('.xlsx'):
        df = pd.read_excel(file_path)
    else:
        print("Unsupported file type.")
        return

    # Print headers and first 10 rows
    print("\nFile Headers:")
    print(df.columns.tolist())
    print("\nFirst 10 Rows:")
    print(df.head(10))

    # Basic AI analysis using OpenAI GPT
    try:
        openai.api_key = api_key
        data_sample = df.head(20).to_string()
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",
            messages=[
                {"role": "system", "content": "You are a data analysis
assistant."},
                {"role": "user", "content": f"Analyze the following
dataset:\n\n{data_sample}\n\nProvide a summary of its contents and structure."}
            ],
```

```
            max_tokens=500
        )
        analysis = response['choices'][0]['message']['content'].strip()
    except Exception as e:
        print(f"Error using OpenAI API: {e}")
        return

    print("\nAI Analysis:")
    print(analysis)

    # Save to Word document
    doc_name =
f"{os.path.splitext(os.path.basename(file_path))[0][:6]}_Dataset_Analysis.docx"
    doc = Document()
    doc.add_heading("Dataset Analysis", level=1)
    doc.add_heading("File Headers", level=2)
    doc.add_paragraph(', '.join(df.columns.tolist()))
    doc.add_heading("First 10 Rows", level=2)
    doc.add_paragraph(df.head(10).to_string())
    doc.add_heading("AI Analysis", level=2)
    doc.add_paragraph(analysis)
    doc.save(doc_name)
    print(f"\nAnalysis saved as: {doc_name}")

def main():
    """
    Main function to prompt for files and perform analysis.
    """
    api_key = ask_api_key()
    while True:
        print("\nFiles in the current directory:")
        for i, file in enumerate(os.listdir(), start=1):
            print(f"{i}. {file}")
        choice = input("Enter the file name to analyze (or type 'exit' to quit):
")
        if choice.lower() == 'exit':
            break
        if not os.path.exists(choice):
            print("File not found. Please try again.")
            continue
        analyze_file(choice, api_key)

if __name__ == "__main__":
    main()
```