**Assignment**

**CSA0814 – Python Programming**

|  |  |
| --- | --- |
| **Register Number** | **192324262** |
| **Name** | **B. Ramya sri** |

**Title:**

Word Document Generator

**Problem Statement:**

Write a Python program that generates Word documents (.docx) using the python-docx module, populating them with data from CSV files, databases, or user input, and applying styling and formatting as required.

**Code:**

import csv

import pandas as pd

from docx import Document

from docx.shared import Pt

from docx.enum.text import WD\_ALIGN\_PARAGRAPH

from docx.shared import Inches

def read\_csv\_data(csv\_file):

"""Read data from a CSV file and return as a DataFrame."""

data = pd.read\_csv(csv\_file)

return data

def create\_word\_document(data, output\_file):

"""Generate a Word document from DataFrame data."""

document = Document()

# Add a title to the document

title = document.add\_heading('Generated Word Document', level=1)

title.alignment = WD\_ALIGN\_PARAGRAPH.CENTER

# Add a table with the data from the DataFrame

table = document.add\_table(rows=1, cols=len(data.columns))

table.style = 'Table Grid'

# Add the header row

hdr\_cells = table.rows[0].cells

for i, column\_name in enumerate(data.columns):

hdr\_cells[i].text = column\_name

hdr\_cells[i].paragraphs[0].runs[0].font.bold = True

hdr\_cells[i].paragraphs[0].alignment = WD\_ALIGN\_PARAGRAPH.CENTER

# Add data rows

for index, row in data.iterrows():

row\_cells = table.add\_row().cells

for i, item in enumerate(row):

row\_cells[i].text = str(item)

# Add some styling

for paragraph in document.paragraphs:

for run in paragraph.runs:

run.font.size = Pt(12)

run.font.name = 'Arial'

# Add a footer to the document

footer = document.sections[0].footer

footer\_paragraph = footer.paragraphs[0]

footer\_paragraph.text = 'Generated by Python-docx'

footer\_paragraph.alignment = WD\_ALIGN\_PARAGRAPH.CENTER

# Save the document

document.save(output\_file)

print(f'Document "{output\_file}" has been created successfully!')

if \_\_name\_\_ == '\_\_main\_\_':

# Example usage:

csv\_file = 'data.csv' # Path to your CSV file

output\_file = 'output.docx' # Output Word document file name

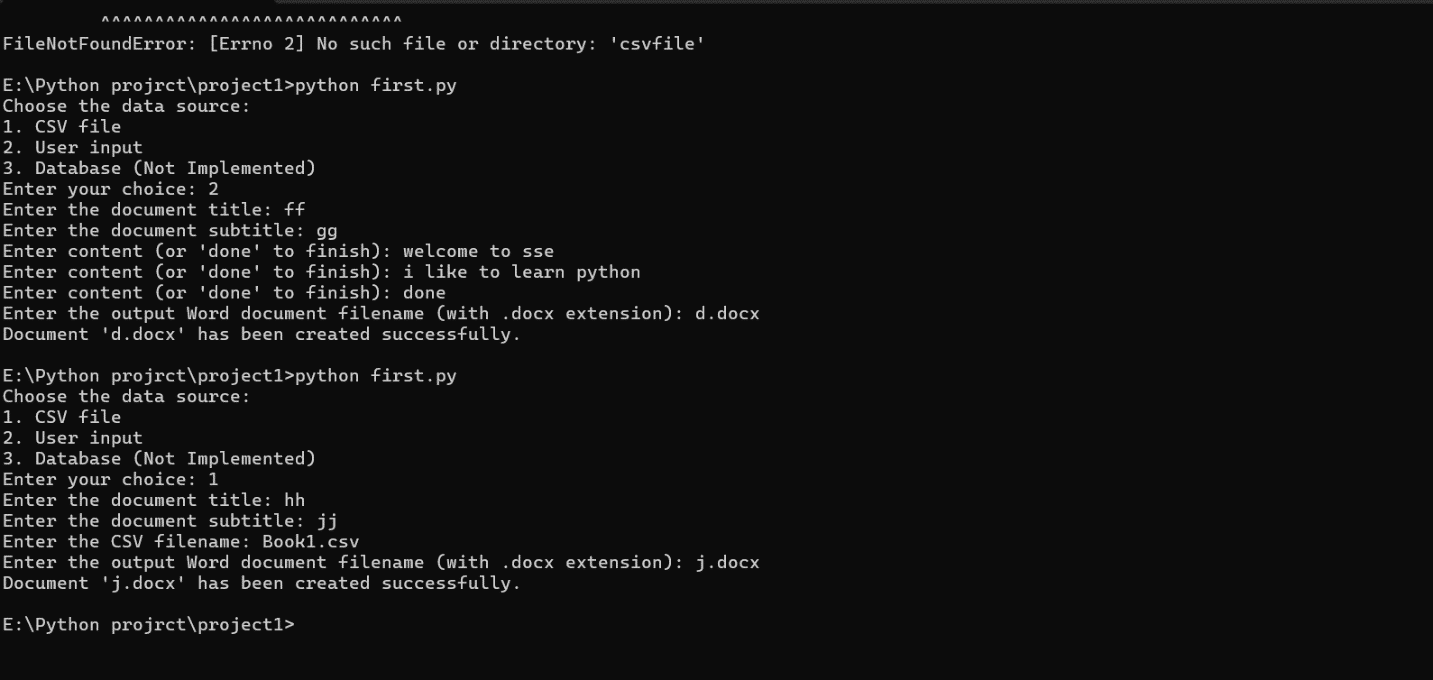
# Read data from the CSV file

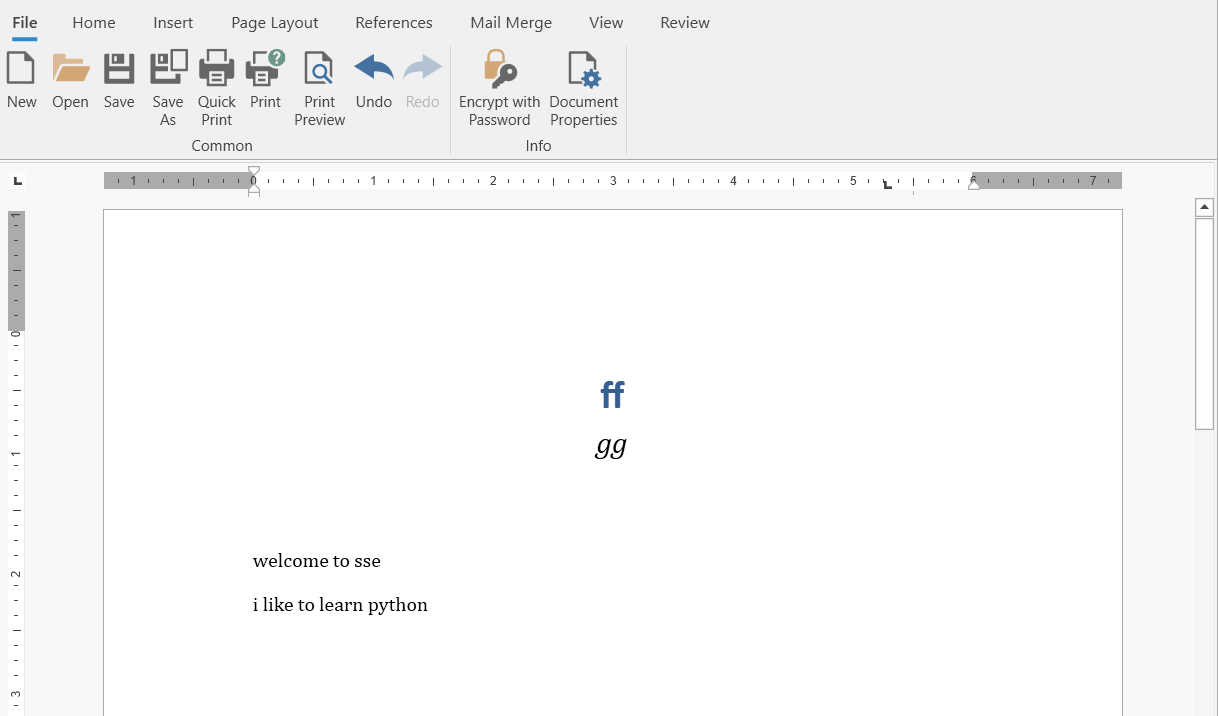
data = read\_csv\_data(csv\_file)

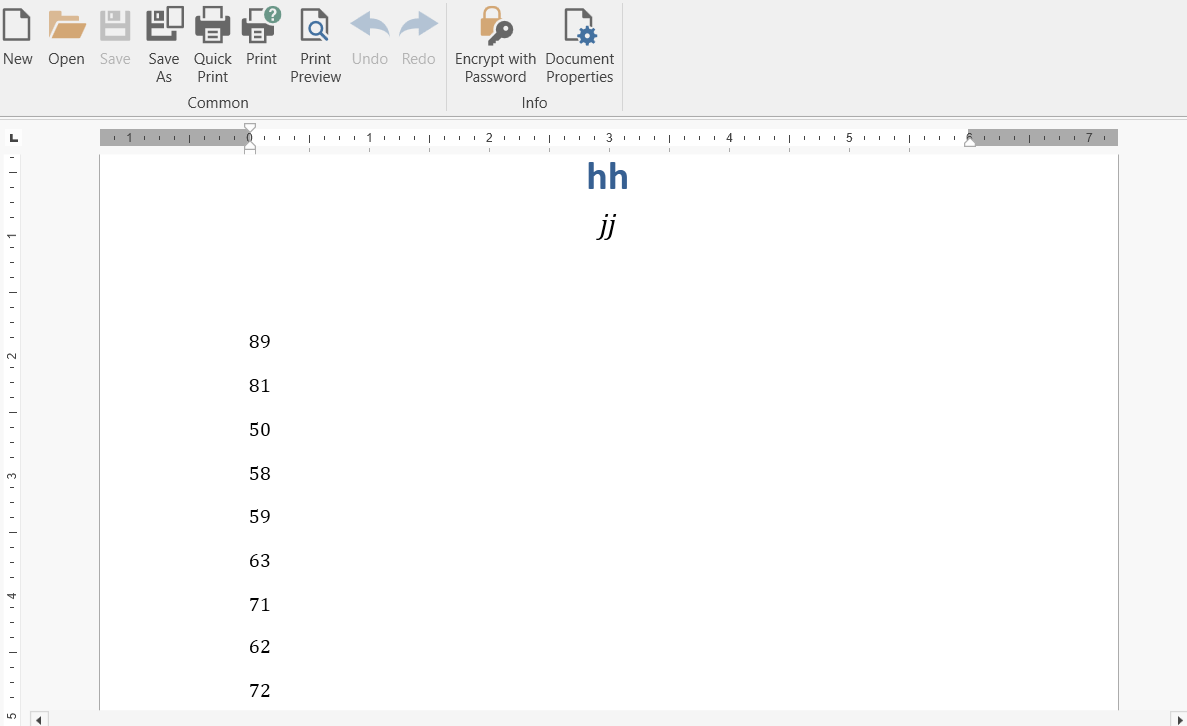
# Create the Word document

create\_word\_document(data, output\_file)

**Output Screen Shots:**

****

****

****

**Conclusion:**

**1. Install Required Packages**

Before executing the script, you need to install the necessary packages:

pip install python-docx pandas

This Python program successfully reads data from a CSV file, applies basic formatting and styling, and generates a Word document using the python-docx module. The process is straightforward, and the program can be easily adapted to pull data from different sources like databases or user input. By leveraging the python-docx API, various customizations and formatting options can be applied to meet specific requirements.

The final Word document is a neatly formatted file that reflects the data from your source, complete with titles, tables, and footers, ready for use or further editing.