# Mastering Excel Based Queries: Efficient Cataloging with Customized Search Tools

SUSIE O'CONNOR
ACCESS SERVICES AND
GOVERNMENT INFORMATION LIBRARIAN
GREGG-GRANITEVILLE LIBRARY AT USC AIKEN

U of SC Aiken
Gregg-Graniteville
Library

# Objective

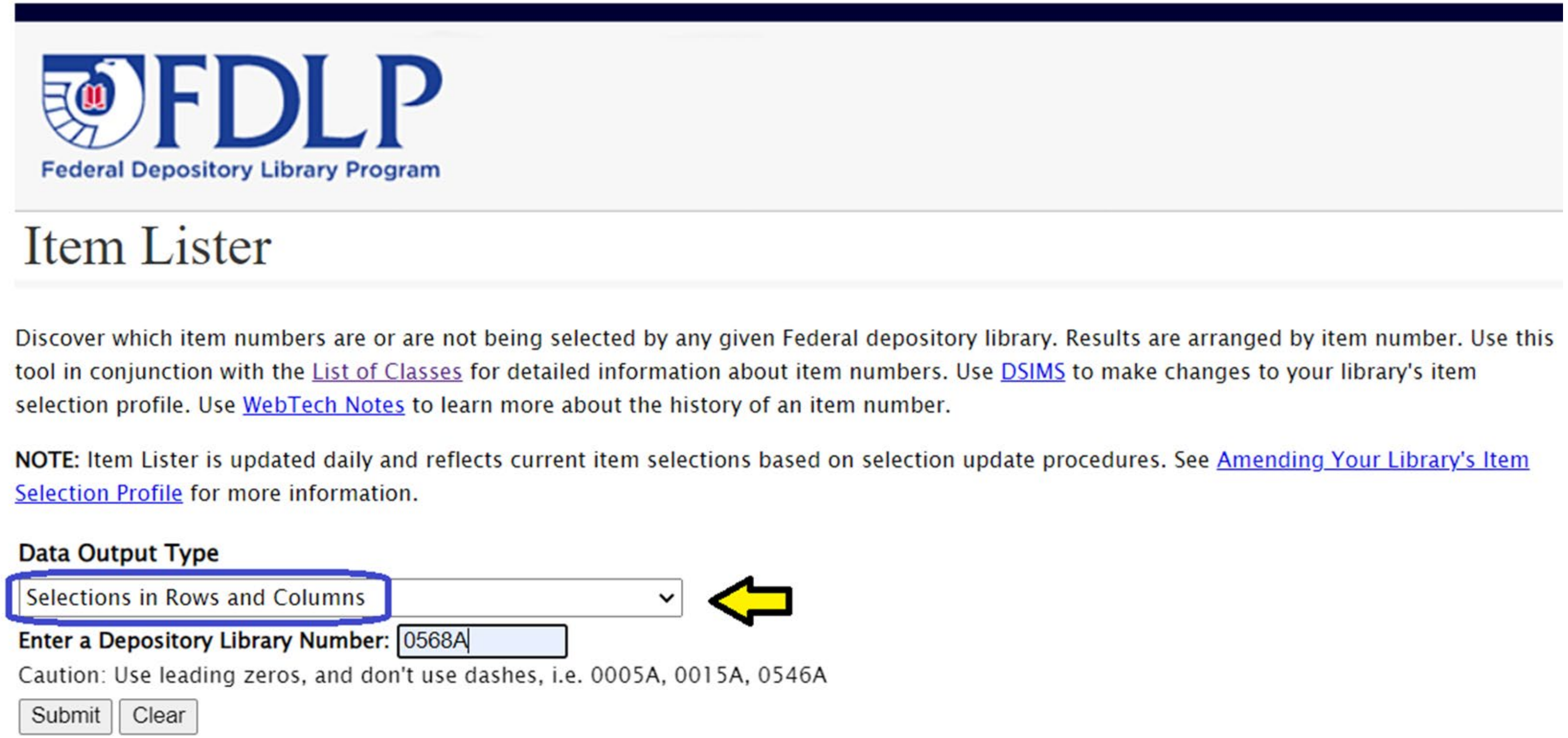
Find a tool that can query a spreadsheet for multiple keywords/terms simultaneously.

**Ingredients:**

1. Criteria/Terms/Keywords
2. Excel file of records
3. Tool/Technology that allows simultaneous keyword searches

# Cataloging Criteria:

*Depository Selections can by located in the Item Lister*



## Item Lister

Discover which item numbers are or are not being selected by any given Federal depository library. Results are arranged by item number. Use this tool in conjunction with the List of Classes for detailed information about item numbers. Use DSIMS to make changes to your library's item selection profile. Use WebTech Notes to learn more about the history of an item number.

**NOTE:** Item Lister is updated daily and reflects current item selections based on selection update procedures. See Amending Your Library's Item Selection Profile for more information.

**Data Output Type**

Selections in Rows and Columns

**Enter a Depository Library Number:** 0568A

Caution: Use leading zeros, and don't use dashes, i.e. 0005A, 0015A, 0546A

# Using Keywords and SuDoc Numbers for Library Collection Searches

## SuDoc Numbers:

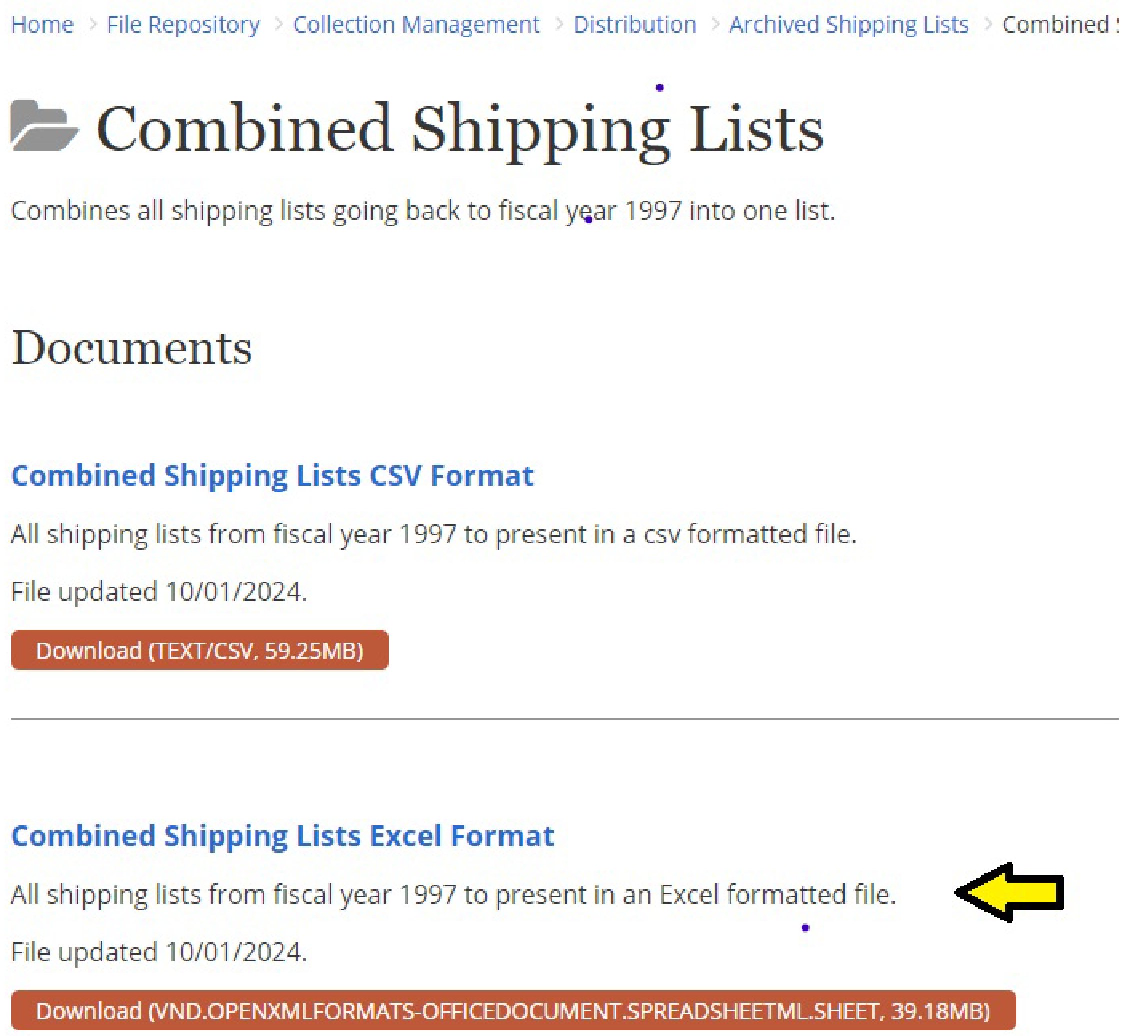
- **Refine your search**: Use SuDoc numbers if you want to avoid retrieving everything under a single item number.

## Keyword Search:

- Target specific library collection areas with relevant keywords
  - Dependent on individual library collection development and teaching areas

Library # 0568A

0001-A-01
0004-B
0004-C
0006
0006-J-03
0006-J-04
0006-J-05
0006-J-06
0006-J-08
0010-A
0010-D
0010-E
0011-F-01
0015
0015-A-03
0015-A-05
0015-A-07
0015-A-08
0015-A-09
0015-A-10
0015-A-12
0015-A-13
0015-A-14
0015-A-15
0015-A-17
0015-A-18
0015-A-19
0015-A-20
0015-A-22
0015-A-23
0015-A-24
0015-A-25
0015-A-27
0015-A-28
0015-A-29
0015-A-30
0015-A-32
0015-A-33
0015-A-34
0015-A-35
0015-A-37
0015-A-38
0015-A-39
0015-A-40
0015-A-42
0015-A-43
0015-A-44
0015-A-45
0015-A-47
0015-A-48
0015-A-49
0015-A-50



Home > File Repository > Collection Management > Distribution > Archived Shipping Lists > Combined !

# Combined Shipping Lists

Combines all shipping lists going back to fiscal year 1997 into one list.

## Documents

### Combined Shipping Lists CSV Format

All shipping lists from fiscal year 1997 to present in a csv formatted file.

File updated 10/01/2024.

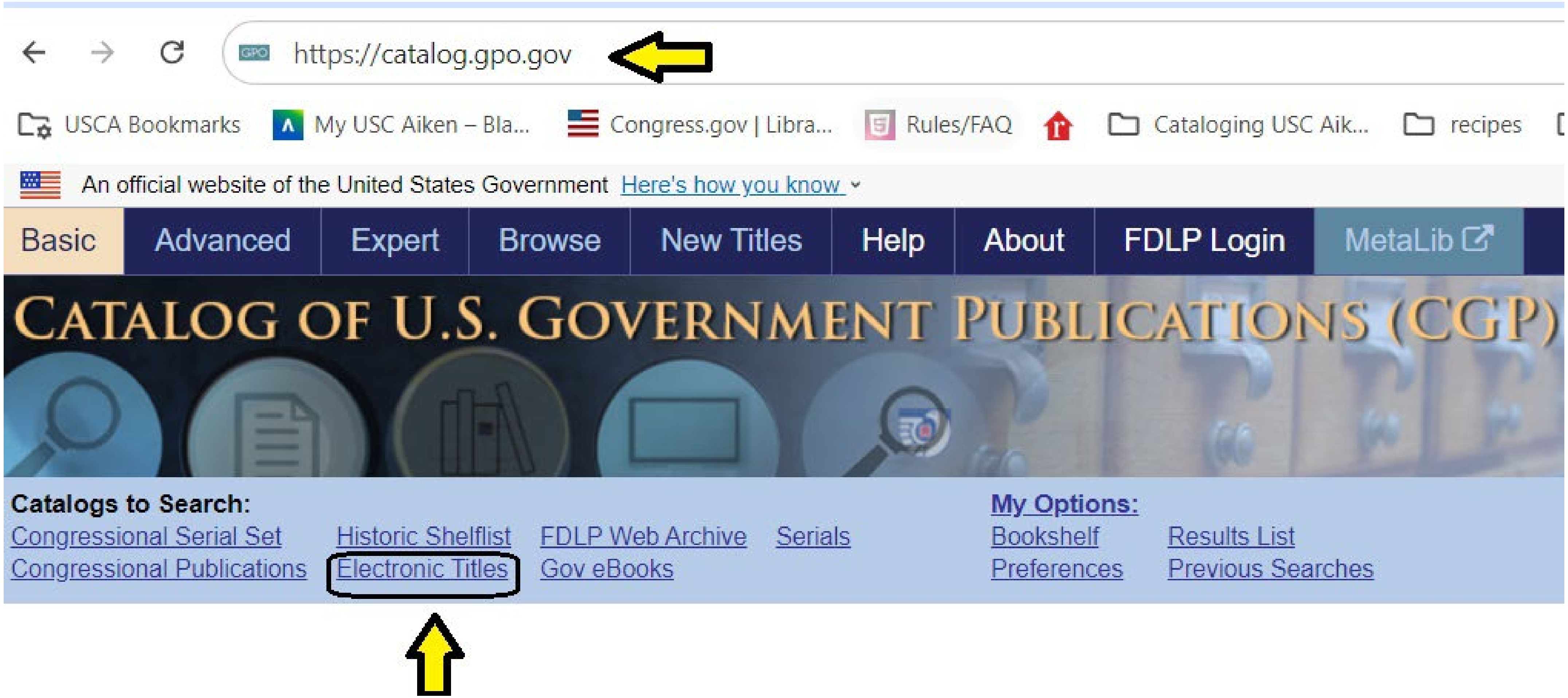
Download (TEXT/CSV, 59.25MB)

### Combined Shipping Lists Excel Format

All shipping lists from fiscal year 1997 to present in an Excel formatted file.

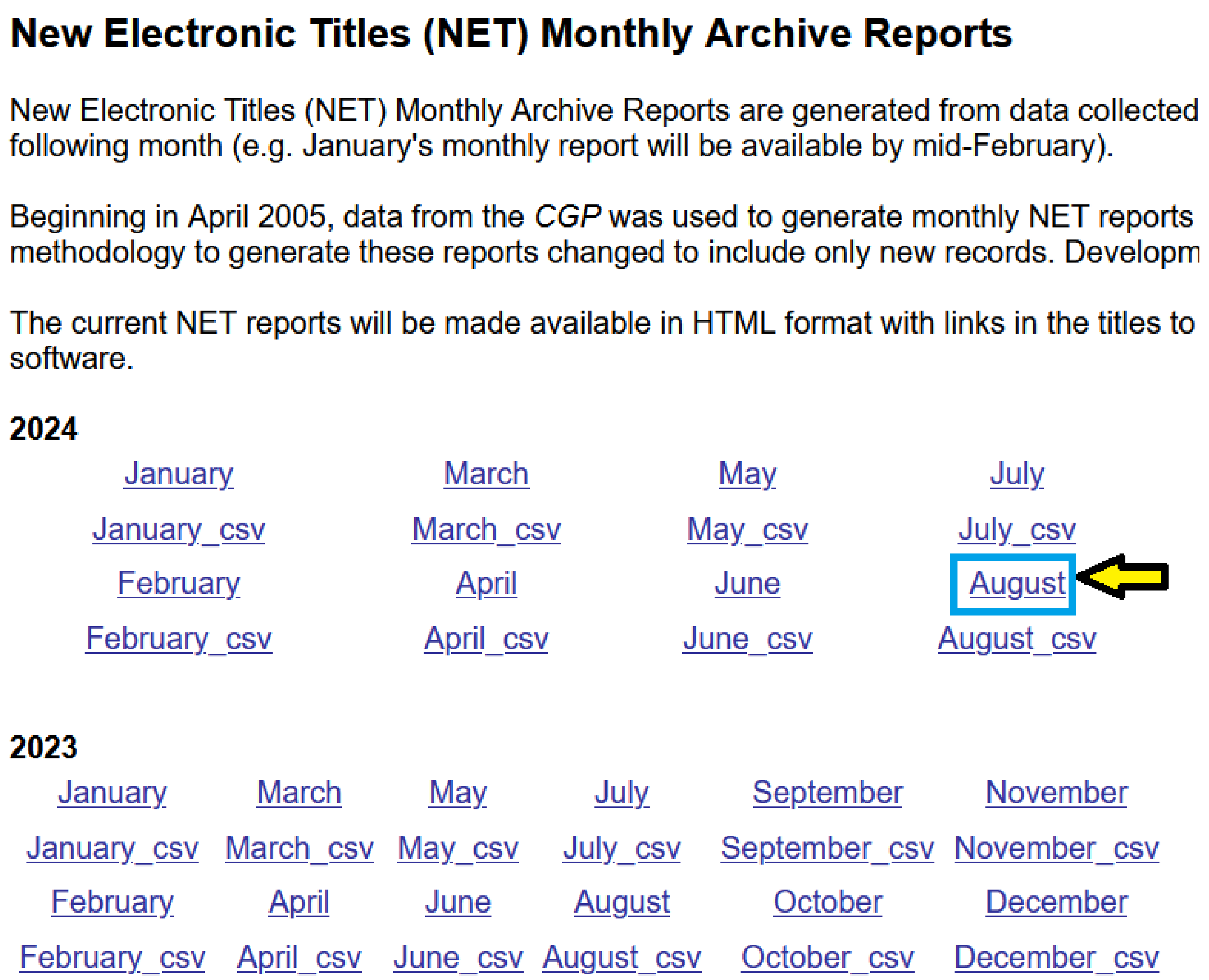
File updated 10/01/2024.

Download (VND.OPENXMLFORMATS-OFFICEDOCUMENT.SPREADSHEETML.SHEET, 39.18MB)



https://catalog.gpo.gov

An official website of the United States Government Here's how you know



## New Electronic Titles (NET) Monthly Archive Reports

New Electronic Titles (NET) Monthly Archive Reports are generated from data collected
following month (e.g. January's monthly report will be available by mid-February).

Beginning in April 2005, data from the *CGP* was used to generate monthly NET reports
methodology to generate these reports changed to include only new records. Developm

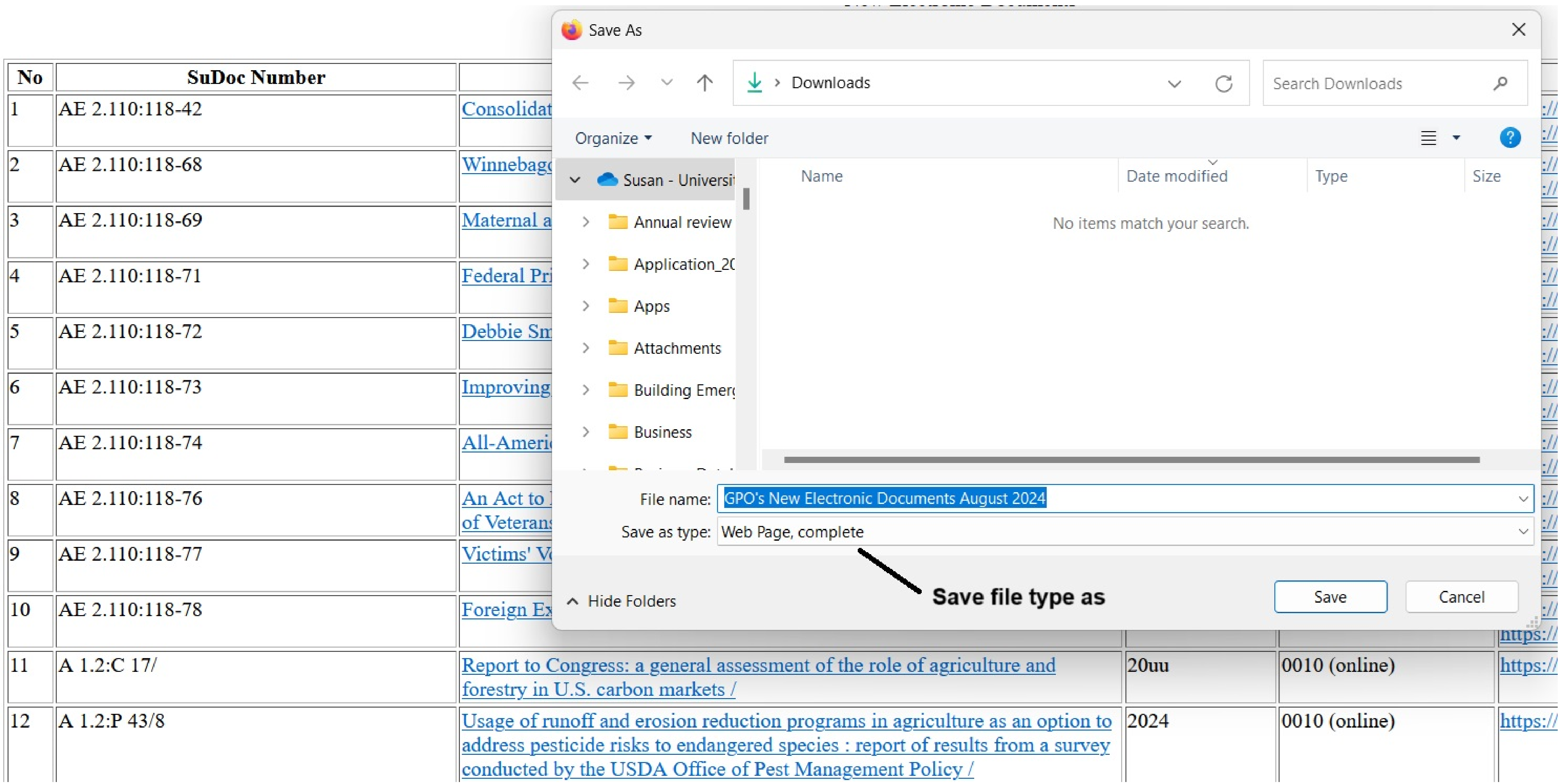
The current NET reports will be made available in HTML format with links in the titles to
software.

**2024**

| | | | |
|---|---|---|---|
| January | March | May | July |
| January_csv | March_csv | May_csv | July_csv |
| February | April | June | August |
| February_csv | April_csv | June_csv | August_csv |

**2023**

| | | | | | |
|---|---|---|---|---|---|
| January | March | May | July | September | November |
| January_csv | March_csv | May_csv | July_csv | September_csv | November_csv |
| February | April | June | August | October | December |
| February_csv | April_csv | June_csv | August_csv | October_csv | December_csv |



| No | SuDoc Number | | | | |
|---|---|---|---|---|---|
| 1 | AE 2.110:118-42 | Consolidat | | | :// |
| 2 | AE 2.110:118-68 | Winnebag | | | :// |
| 3 | AE 2.110:118-69 | Maternal a | | | :// |
| 4 | AE 2.110:118-71 | Federal Pri | | | :// |
| 5 | AE 2.110:118-72 | Debbie Sm | | | :// |
| 6 | AE 2.110:118-73 | Improving | | | :// |
| 7 | AE 2.110:118-74 | All-Ameri | | | :// |
| 8 | AE 2.110:118-76 | An Act to I of Veterans | | | :// |
| 9 | AE 2.110:118-77 | Victims' V | | | :// |
| 10 | AE 2.110:118-78 | Foreign Ex | | | https:// |
| 11 | A 1.2:C 17/ | Report to Congress: a general assessment of the role of agriculture and forestry in U.S. carbon markets / | 20uu | 0010 (online) | https:// |
| 12 | A 1.2:P 43/8 | Usage of runoff and erosion reduction programs in agriculture as an option to address pesticide risks to endangered species : report of results from a survey conducted by the USDA Office of Pest Management Policy / | 2024 | 0010 (online) | https:// |

Save As

No items match your search.

File name: GPO's New Electronic Documents August 2024

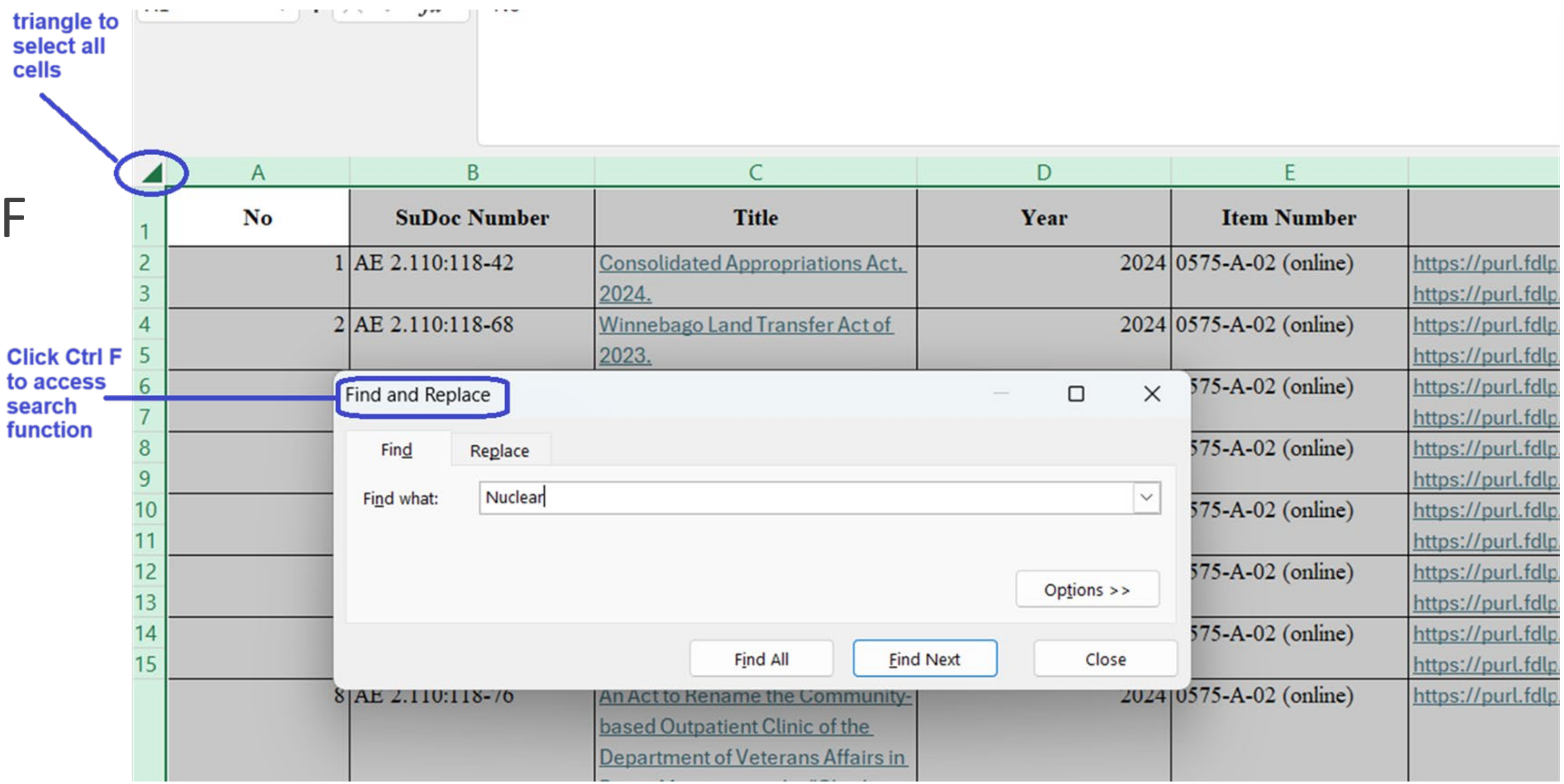
Save as type: Web Page, complete

Save file type as

# Options for Advanced Searching:

## *Easy/Beginner Computer Knowledge*

Excel and Ctrl F



triangle to select all cells

Click Ctrl F to access search function

| No | SuDoc Number | Title | Year | Item Number | |
|---|---|---|---|---|---|
| 1 | AE 2.110:118-42 | Consolidated Appropriations Act, 2024. | 2024 | 0575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| 2 | AE 2.110:118-68 | Winnebago Land Transfer Act of 2023. | 2024 | 0575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| | | | | 575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| | | | | 575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| | | | | 575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| | | | | 575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| | | | | 575-A-02 (online) | https://purl.fdlp<br>https://purl.fdlp |
| 8 | AE 2.110:118-76 | An Act to Rename the Community-based Outpatient Clinic of the Department of Veterans Affairs in | 2024 | 0575-A-02 (online) | https://purl.fdlp |

Find and Replace

Find what: Nuclear

# Options for Advanced Searching:

*Intermediate to advanced Computer Knowledge*

- Python Script with PyCharm
  - Python is a programming language
- AI Tools and extensions
- Microsoft Power Automate
- Power Query (Excel Extension)

# Python Code Development for Automated Data Search

**Objective:**

Python code was developed to streamline data searches and automate results extraction, following these steps:

**1. Define the Search Terms:**

A list of terms in a .txt file was used to guide this process

**2. Retrieve Matching Entries:**

The code searches an Excel workbook and identifies all entries that exactly match the defined terms.

**3. Organize Results:**

All matching entries are collected and placed into a separate Excel sheet for easier review and further analysis.

# Tools and Setup for Python Coding

**Tools Used:**

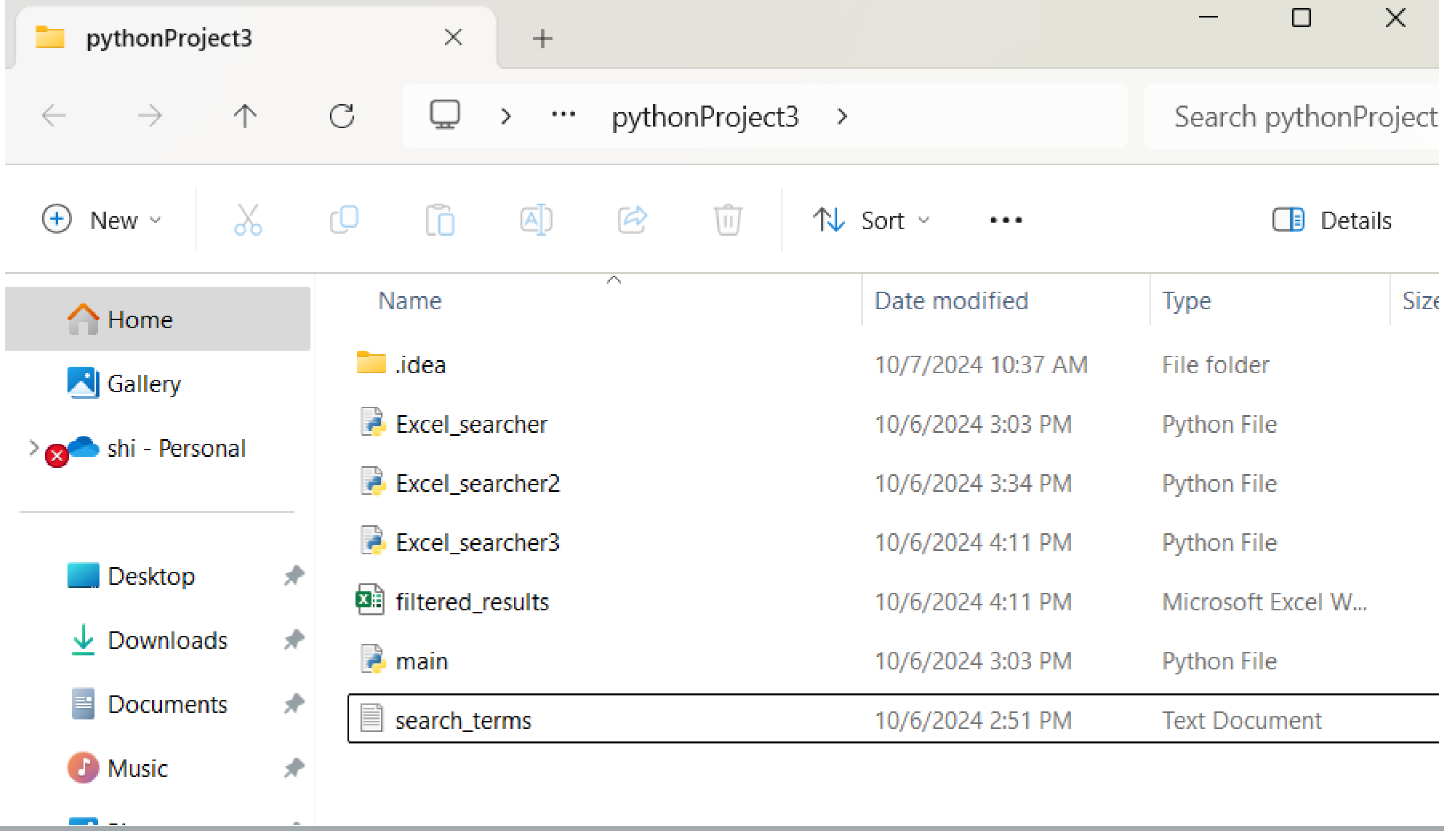
1. **PyCharm**
   1. One of many available IDEs for Python coding. Used for this project.
2. **Pandas Module**
   1. A Python library essential for data manipulation and analysis.
3. **Openpyxl Module**
   1. Required by Pandas to read from and write to Excel files.

**Setup Instructions:**

- **Run PyCharm in Administrator Mode:**
  - Right-click the PyCharm icon and select "Run as Administrator."
  - **Note**: Administrator mode is necessary for Python to read and write files.

# Set-up part 2

*It is important that ALL files are set up in the same folder



| Name | Date modified | Type | Size |
|---|---|---|---|
| .idea | 10/7/2024 10:37 AM | File folder | |
| Excel_searcher | 10/6/2024 3:03 PM | Python File | |
| Excel_searcher2 | 10/6/2024 3:34 PM | Python File | |
| Excel_searcher3 | 10/6/2024 4:11 PM | Python File | |
| filtered_results | 10/6/2024 4:11 PM | Microsoft Excel W... | |
| main | 10/6/2024 3:03 PM | Python File | |
| search_terms | 10/6/2024 2:51 PM | Text Document | |

# Key Code Adjustments

**File Path Modification:**

The file path to access the Excel file needed adjustments for proper entry.Text

**String Changes:**

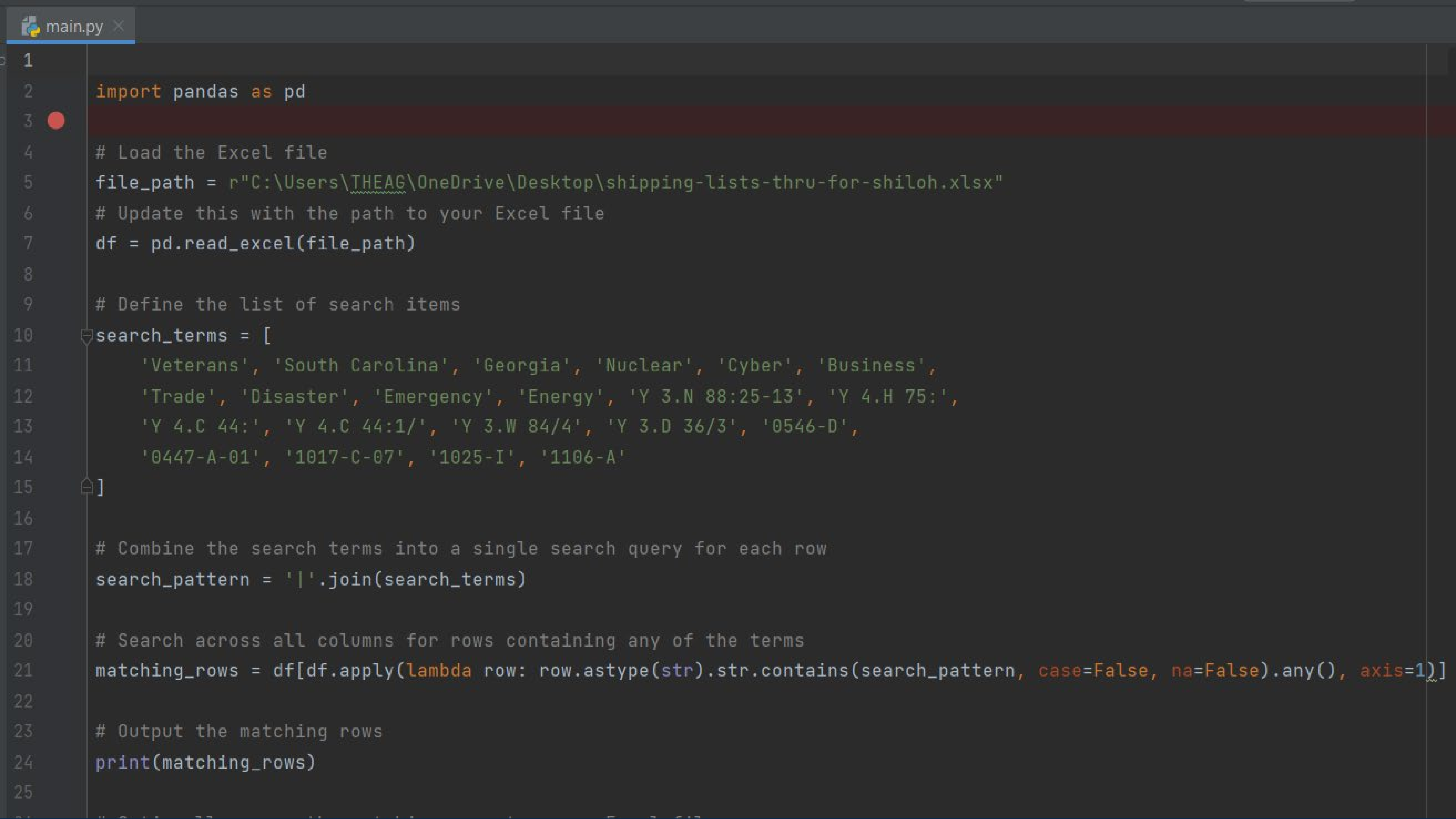
The text was changed to a raw string (r" ") to handle backslashes in the file path correctly.

# Lessons Learned

**Limitations of ChatGPT:**

- **File Path Formatting:**
  - ChatGPT cannot predict or auto-format file paths specific to the user's system.
- **Code Testing:**
  - It is essential to verify that the code runs without errors.
  - Multiple iterations and tests were required to refine the code.

# Early Iteration of Code



```python

import pandas as pd

# Load the Excel file
file_path = r"C:\Users\THEAG\OneDrive\Desktop\shipping-lists-thru-for-shiloh.xlsx"
# Update this with the path to your Excel file
df = pd.read_excel(file_path)

# Define the list of search items
search_terms = [
    'Veterans', 'South Carolina', 'Georgia', 'Nuclear', 'Cyber', 'Business',
    'Trade', 'Disaster', 'Emergency', 'Energy', 'Y 3.N 88:25-13', 'Y 4.H 75:',
    'Y 4.C 44:', 'Y 4.C 44:1/', 'Y 3.W 84/4', 'Y 3.D 36/3', '0546-D',
    '0447-A-01', '1017-C-07', '1025-I', '1106-A'
]

# Combine the search terms into a single search query for each row
search_pattern = '|'.join(search_terms)

# Search across all columns for rows containing any of the terms
matching_rows = df[df.apply(lambda row: row.astype(str).str.contains(search_pattern, case=False, na=False).any(), axis=1)]

# Output the matching rows
print(matching_rows)

```

# Limitations of ChatGPT in Task Execution

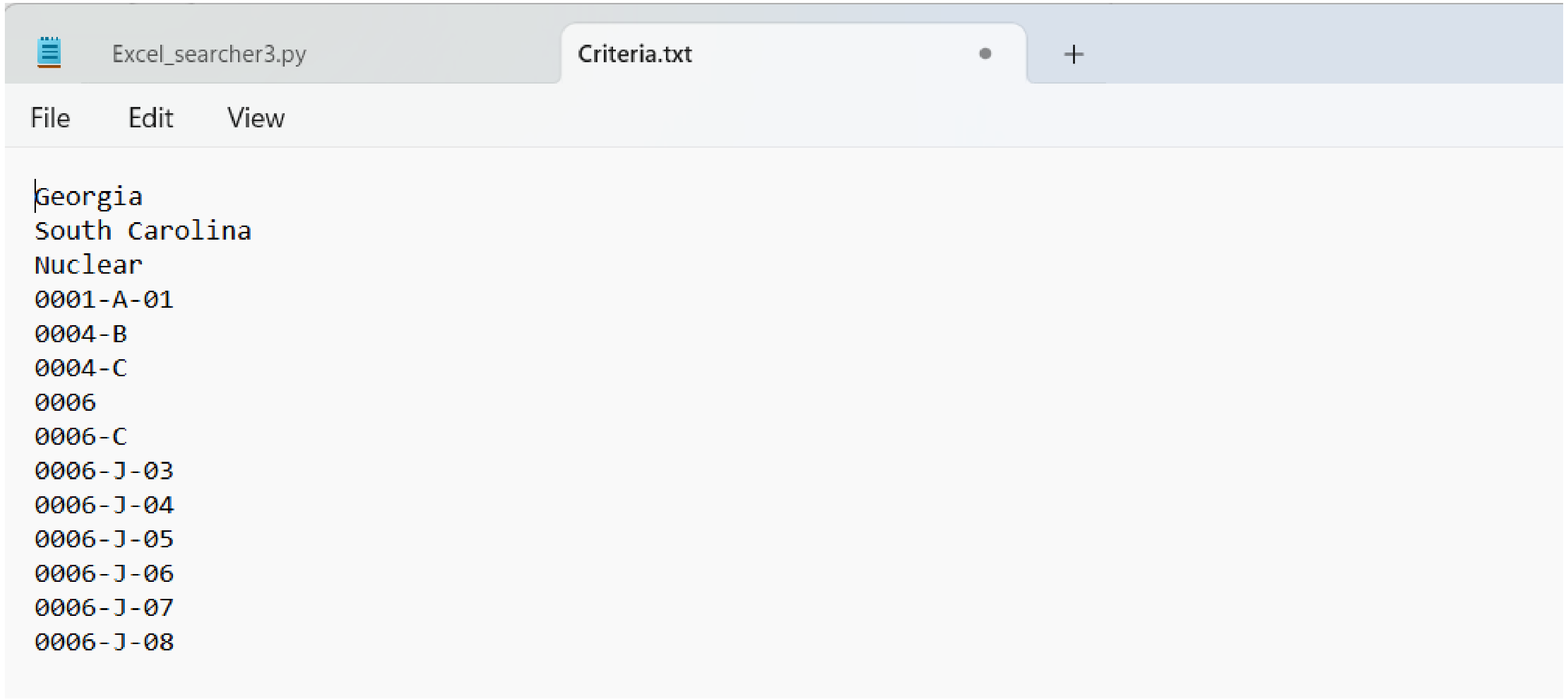
- ChatGPT is a computer-based tool; it gives you exactly what you ask for and cannot anticipate problems or situations that are unforeseen by the user.

**Supporting Examples:**

- **Efficiency Issue**: Using a text file can be faster than manually typing keywords.
- **Search Refinement**: It took several iterations to refine a search for depository selections.
    - *Initial Mismatch*: Search first pulled up "0050-E-17," which was incorrect. The tool matched on the partial number "0050-E" from my actual selection, causing incorrect results.
    - *Lesson Learned*: The code ChatGPT provided matched anything with the same prefix, leading to results that did not fit the specific needs. Changes to the code to make an exact match was required.

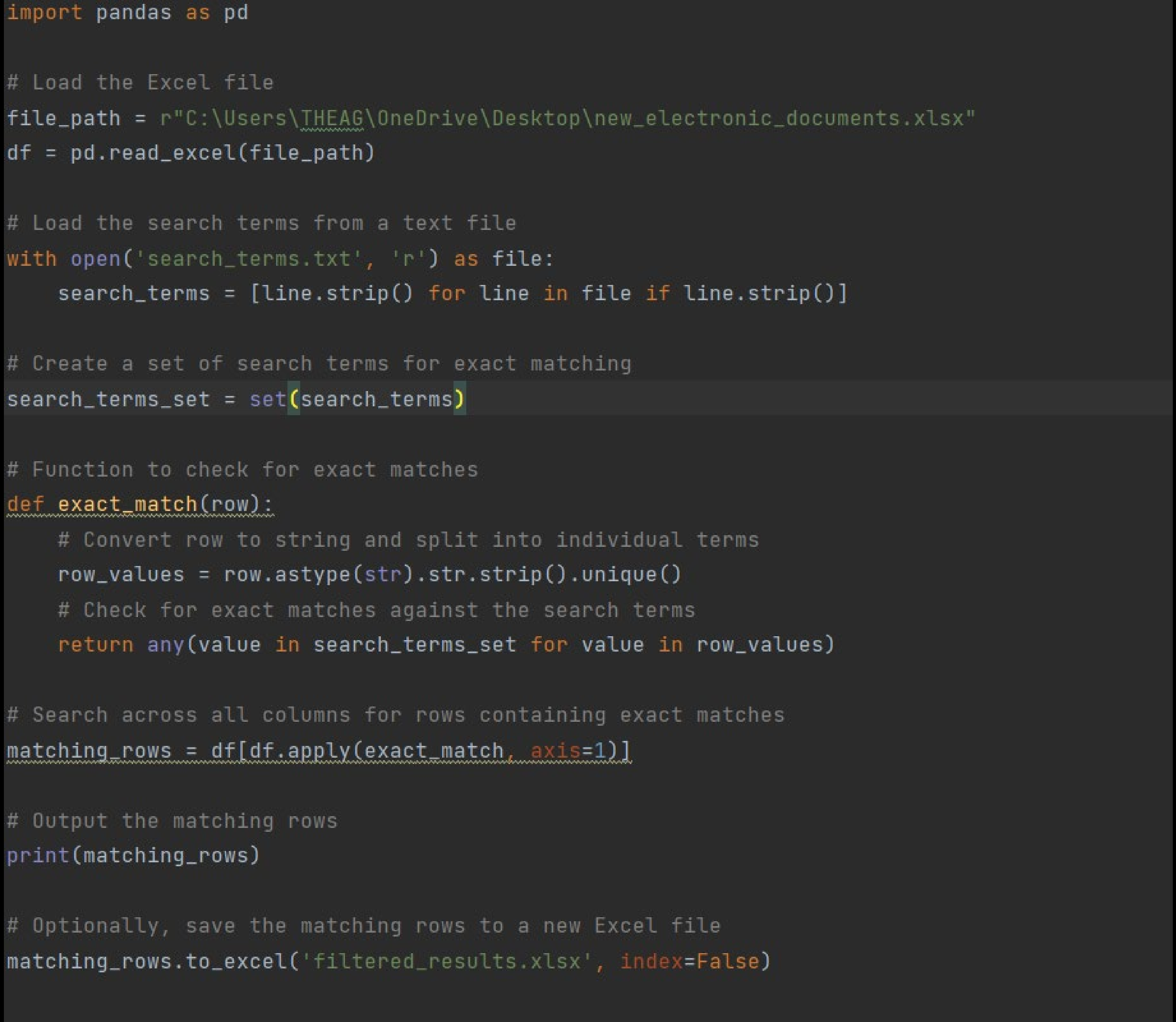
# Notepad Tips

- Make sure there is no space at the top of the sheet
- Words on the same line are treated as one search term



```
Georgia
South Carolina
Nuclear
0001-A-01
0004-B
0004-C
0006
0006-C
0006-J-03
0006-J-04
0006-J-05
0006-J-06
0006-J-07
0006-J-08
```

# Final Coding Product



```python
import pandas as pd

# Load the Excel file
file_path = r"C:\Users\THEAG\OneDrive\Desktop\new_electronic_documents.xlsx"
df = pd.read_excel(file_path)

# Load the search terms from a text file
with open('search_terms.txt', 'r') as file:
    search_terms = [line.strip() for line in file if line.strip()]

# Create a set of search terms for exact matching
search_terms_set = set(search_terms)

# Function to check for exact matches
def exact_match(row):
    # Convert row to string and split into individual terms
    row_values = row.astype(str).str.strip().unique()
    # Check for exact matches against the search terms
    return any(value in search_terms_set for value in row_values)

# Search across all columns for rows containing exact matches
matching_rows = df[df.apply(exact_match, axis=1)]

# Output the matching rows
print(matching_rows)

# Optionally, save the matching rows to a new Excel file
matching_rows.to_excel('filtered_results.xlsx', index=False)
```

# Credits

**Credits:**

- OpenAI (ChatGPT)
  *Personal communication, October, 2024*
- Shiloh O’Connor
  *Assistance with Python code*

A