Tagle, Marc Neil V.
M001

CP102 Midterm Output
Documentation

Part I: CSV File

Instructions: Create a record management system program that displays a menu that will let the user select options: Add new record, Display all records, and Search record.

Optional Requirement: Include Delete record functionality, Implement Object-Oriented Programming (OOP).

CSV File Manager

This program provides a simple interface to manage CSV files using Python.

The program allows users to:

- View the contents of a CSV file.
- Search for specific data in a chosen column.
- Add new rows of data.
- Delete rows based on a search value.
- Empty the contents of a CSV file (with or without headers).
- Change the CSV file being managed.

The program utilizes the following modules:

- **csv** – For reading, writing, and modifying CSV files.
- **tabulate** – For displaying tabular data in a well-formatted way.
- **os** – To check if the specified file exists before accessing it.

The program ensures error handling by validating user input and preventing crashes due to missing files or incorrect data entries.

menu.py

```
# Tagle, Marc Neil V. (M001)

import os
from csv_file_manager import CSV_Manager

def get_valid_file():
    while True:
        file_name = input("Enter file name (with .csv extension): ").strip()
        if not file_name.endswith(".csv"):
            print("File is not CSV.") # Prints if file is not a CSV file.
            continue
        if not os.path.isfile(file_name):
            print(f"File not found.") # Prints if file is not found.
            continue
        return file_name # Returns the valid file name for use.

def separator():
    print("=" * 39)

def goodbye():
    print("Exiting CSV Manager. Goodbye!")

try:
    if __name__ == '__main__':
        print("=" * 13, "CSV MANAGER",  "=" * 13)
        file_name = get_valid_file() # Prompts user to enter a valid file name.
        manager = CSV_Manager(file_name) # Creates an object from 'CSV_Manager' class with the specified file name.

        proceed = True
        while proceed:
            separator()
            print("CSV Manager:")
            print("1 - View\n2 - Search\n3 - Add\n4 - Delete Row/s\n5 - Empty Out\n6 - Change File\n7 - Exit")

            try:
                operation = int(input("Enter operation: "))
            except ValueError: # Captures exception when user enters a non-numeric operation.
                print("Invalid operation!")
                continue
            separator()

            if operation == 1:
                manager.view_csv()

            elif operation == 2:
                manager.search_csv()

            elif operation == 3:
                manager.add_row()

            elif operation == 4:
                manager.delete_row()

            elif operation == 5:
                manager.empty_out()

            elif operation == 6:
                new_file = get_valid_file()
                manager.change_file(new_file)

            elif operation == 7:
                goodbye()
                break

            else: # Prints if user enters a invalid numeric operation.
                print("Invalid operation!")
                continue

            while True:
                cont = input("\nDo you want to perform another operation (y/n): ").lower()
                if cont in ['y','n']:
                    break
                print("Invalid response! Enter 'y' or 'n'.")

            if cont == 'n':
                proceed = False
                separator()
                goodbye()

except Exception as e: # Captures any unexpected errors to avoid crashes.
    print(f"An unexpected error occurred: {e}")
```

csv_file_manager.py

```
# Tagle, Marc Neil V. (M001)

import csv
from tabulate import tabulate # Tabulate is a third-party module

class CSV_Manager:
    def __init__(self, file_name):
        self.file_name = file_name # Created object accepts a file name as argument.

    def view_csv(self):
        with open(self.file_name, 'r', newline = '') as csv_file:
            reader = csv.reader(csv_file)
            rows = list(reader) # Creates a list of all rows from the 'reader' iterator.

        if not rows: # Checks if file is empty.
            print("This file is empty.")
            return

        print(f"Processed {len(rows) - 1} lines (excluding headers):")
        if len(rows[0]) < 8: # Tabulates data if dataset if small.
            print(tabulate(rows, headers="firstrow", tablefmt="grid"))
        else:
            print("\n",", ".join(rows[0]),"\n")
            if len(rows[1:]) < 50: # If number of rows is less than 50, it shows all rows.
                for row in rows[1:]:
                    print("\t", ", ".join(row))
            else: # Shows a preview of the dataset if number of rows is greater than 50.
                for row in rows[1:6]:
                    print("\t", ", ".join(row))
                print("\t...") # Dataset is truncated.
                for row in rows[-5:]:
                    print("\t", ", ".join(row))

    def search_csv(self):
        with open(self.file_name, 'r', newline='') as csv_file:
            reader = csv.DictReader(csv_file)
            rows = list(reader)
            field_names = reader.fieldnames

        if not rows or not field_names: # Checks if there are no rows or no columns.
            print("This file is empty.")
            return

        print("Columns:") # Shows all columns available.
        for field in field_names:
            print(f" - {field}")

        attempts = 3 # Limits number of tries for entering a valid column.
```

```
# If all three attempts fail, the function exits and prompts the user for another operation.
while attempts > 0:
    column = input("\nEnter column (or type 'exit' to cancel): ")
    if column.lower() == 'exit': # User can cancel searching.
        return
    if column not in field_names:
        attempts -= 1 # Number of attempts is decremented upon entering an invalid column.
        print(f"Error: Column {column} not found! ({attempts} attempt(s) left)")
        continue
    else: # Entering a valid column will let user enter a search value.
        search_value = input("Search: ")
        # Filters the list 'rows' to find all rows where the value in the column matches
        # the given 'search_value' and stores them in 'rows_found'.
        rows_found = [row for row in rows if row[column] == search_value]

        if rows_found:
            print(f"{len(rows_found)} row/s found:")
            print("\n",", ".join(field_names),"\n") # Prints field names as header.
            if len(rows_found[0]) < 8: # Tabulates data if dataset if small.
                print(tabulate(rows_found, headers="keys", tablefmt="grid"))
            else: # If dataset has too many columns, searched rows are printed normally line by line.
                for row in rows_found:
                    print("\t", ", ".join(row.values()))
        else:
            print("No match.") # Prints if no rows match.
            break

    def add_row(self):
        with open(self.file_name, 'r', newline='') as csv_file:
            reader = csv.DictReader(csv_file)
            field_names = reader.fieldnames

        if not field_names: # Checks if file is empty.
            print("This file is empty.")
            return

        new_row_list = []
        for field in field_names:
            row_value = input(f"Enter {field}: ")
            if row_value.lower() == 'exit': # User can cancel adding a row.
                return
            new_row_list.append(row_value)
        new_row_dict = {field: new_row_list[i] for i, field in enumerate(field_names)}

        with open(self.file_name, 'a', newline = '') as csv_file:
            writer = csv.DictWriter(csv_file, fieldnames=field_names)
            writer.writerow(new_row_dict) # A new row is added with values of 'new_row_dict'
        print("New row added successfully!")

    def delete_row(self):
```

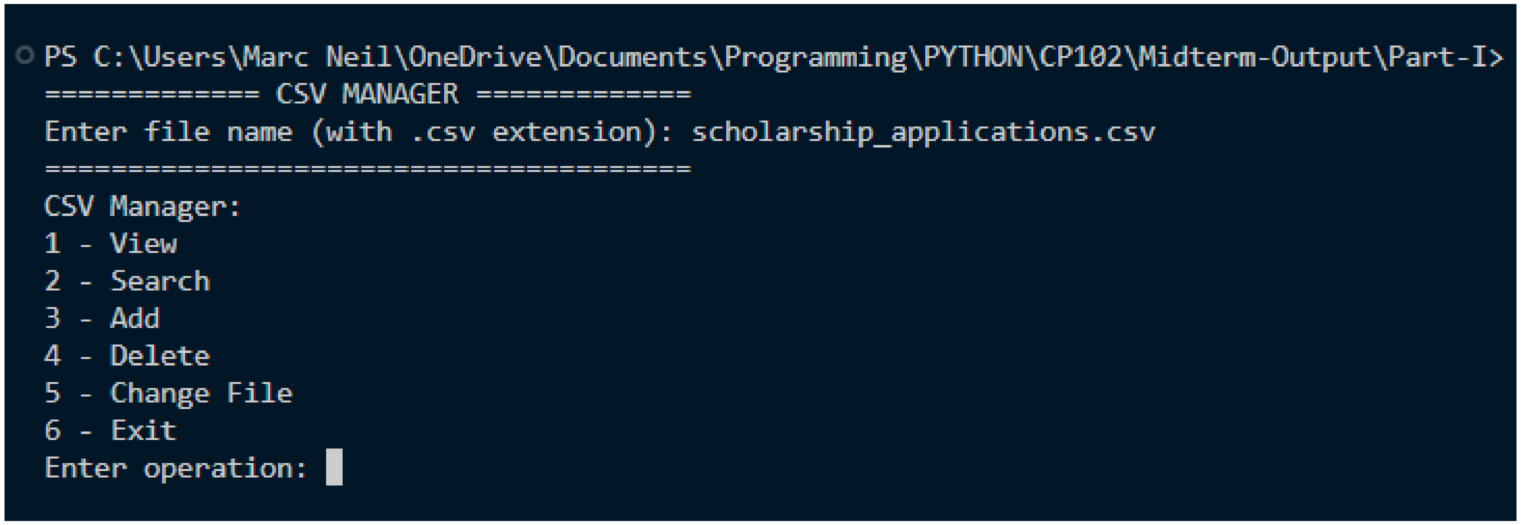
```
with open(self.file_name, 'r', newline = '') as csv_file:
    reader = csv.DictReader(csv_file)
    rows = list(reader)
    field_names = reader.fieldnames

if not rows or not field_names: # Checks if there are no rows or no columns.
    print("This file is empty.")
    return

print("Columns:") # Shows all columns available.
for field in field_names:
    print(f" - {field}")

attempts = 3 # Limits number of tries for entering a valid column.
# If all three attempts fail, the function exits and prompts the user for another operation.
while attempts > 0:
    column = input("\nEnter column (or type 'exit' to cancel): ")
    if column.lower() == 'exit': # User can cancel deleting row/s.
        return
    if column not in field_names:
        attempts -= 1 # Number of attempts is decremented upon entering an invalid column.
        print(f"Error: Column {column} not found! ({attempts} attempt(s) left)")
        continue
    else: # Entering a valid column will let user enter a search value.
        delete_value = input("Delete: ")
```

```
# Creates a list of all rows where the value in the specified 'column'
# matches 'delete_value' (these rows will be deleted).
rows_deleted = [row for row in rows if row[column] == delete_value]

# Creates a list of all rows where the value in the specified 'column'
# does NOT match 'delete_value' (these rows will remain in the dataset).
rows_remain = [row for row in rows if row[column] != delete_value]

# Checks if any rows were deleted.
if len(rows_remain) == len(rows) and not rows_deleted:
    print("No row/s deleted.")
else:
    # Opens the CSV file to overwrite the content with 'rows_remain'.
    with open(self.file_name, 'w', newline = '') as csv_file:
        writer = csv.DictWriter(csv_file, fieldnames=field_names)
        writer.writeheader()
        writer.writerows(rows_remain)

    print(f"{len(rows_deleted)} row/s deleted successfully:") #  Shows deleted rows.
    print("\n",", ".join(field_names),"\n") # Prints field names as header.
    if len(rows_deleted[0]) < 8:
        # Tabulates data if dataset if small.
        print(tabulate(rows_deleted, headers="keys", tablefmt="grid"))
    else:
```

```
                    # If dataset has too many columns, deleted rows are printed normally line by line.
                    for row in rows_deleted:
                        print("\t", ", ".join(row.values()))
            break

def empty_out(self):
    with open(self.file_name, 'r', newline = '') as csv_file:
        reader = csv.DictReader(csv_file)
        rows = list(reader)
        field_names = reader.fieldnames

    if not rows:
        print("This file is already empty or has no headers.")
        return

    while True:
        keep_header = input("Do you want to keep the headers? (y/n): ").lower()
        if keep_header in ['y','n']:
            break
        print("Invalid response! Enter 'y' or 'n'.")

    with open(self.file_name, 'w', newline = '') as csv_file:
        writer = csv.DictWriter(csv_file, fieldnames=field_names)

        if keep_header == 'y':
                writer.writeheader()
                print(f"{self.file_name} has been emptied successfully, but headers were kept.")
            else:
                print(f"{self.file_name} has been completely emptied.")

    # Updates the file name to a new specified file and prints a confirmation message.
    def change_file(self, new_file):
        self.file_name = new_file
        print(f"File changed to {self.file_name}!")
```

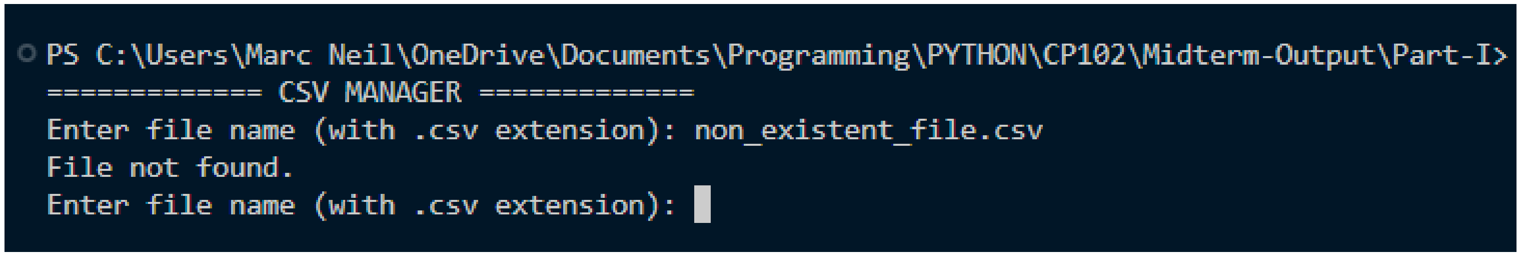
Screenshots of Output

A. Entering file name



```
PS C:\Users\Marc Neil\OneDrive\Documents\Programming\PYTHON\CP102\Midterm-Output\Part-I>
============= CSV MANAGER =============
Enter file name (with .csv extension): scholarship_applications.csv
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 
```

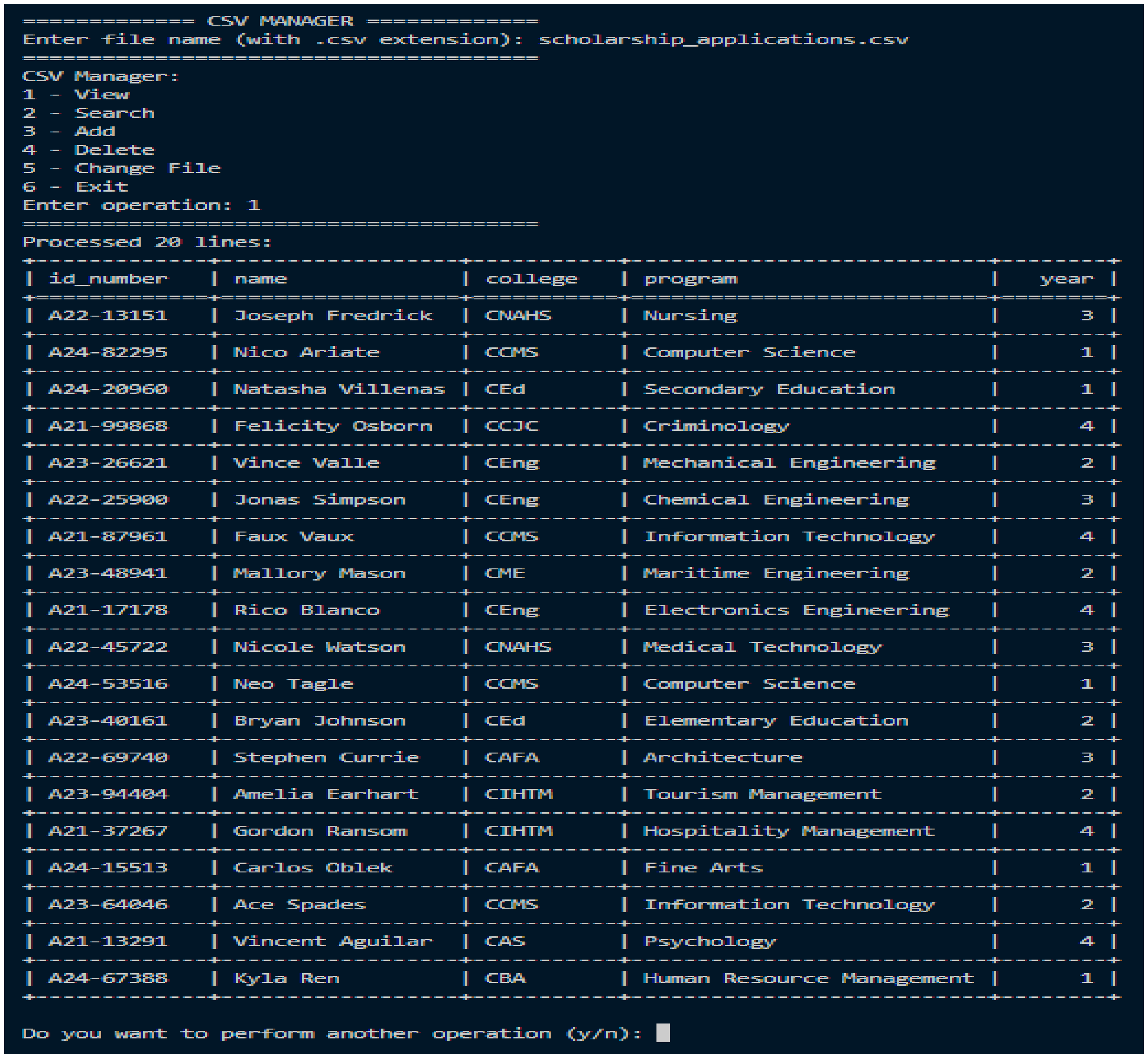
Entering name of an existing file.



```
PS C:\Users\Marc Neil\OneDrive\Documents\Programming\PYTHON\CP102\Midterm-Output\Part-I>
============= CSV MANAGER =============
Enter file name (with .csv extension): non_existent_file.csv
File not found.
Enter file name (with .csv extension): 
```

Entering name of a non-existing file.

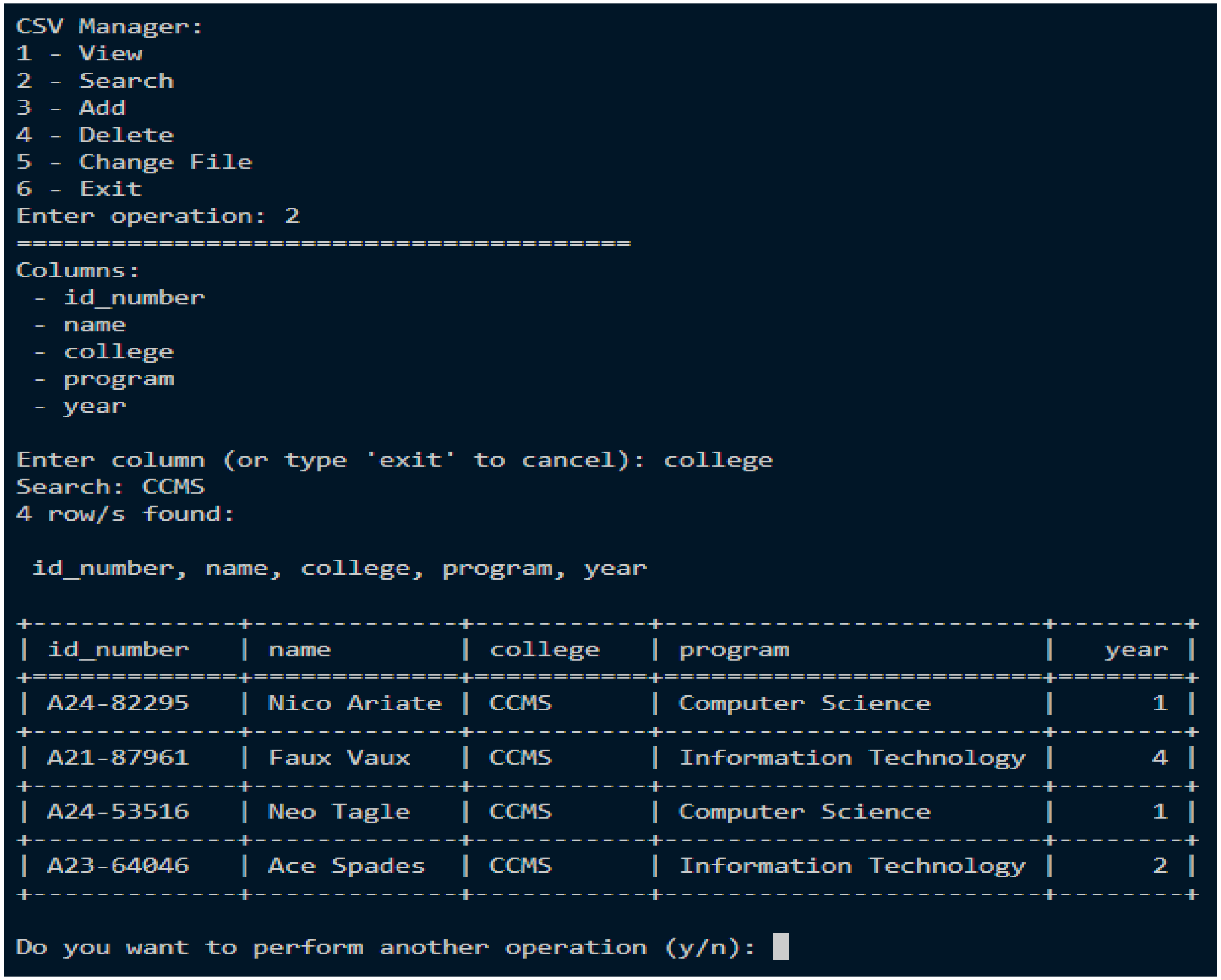
B. Viewing data



```
============= CSV MANAGER =============
Enter file name (with .csv extension): scholarship_applications.csv
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 1
=======================================
Processed 20 lines:
+-------------+------------------+-----------+---------------------------+--------+
| id_number   | name             | college   | program                   |   year |
+=============+==================+===========+===========================+========+
| A22-13151   | Joseph Fredrick  | CNAHS     | Nursing                   |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-82295   | Nico Ariate      | CCMS      | Computer Science          |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-20960   | Natasha Villenas | CEd       | Secondary Education       |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-99868   | Felicity Osborn  | CCJC      | Criminology               |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-26621   | Vince Valle      | CEng      | Mechanical Engineering    |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A22-25900   | Jonas Simpson    | CEng      | Chemical Engineering      |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-87961   | Faux Vaux        | CCMS      | Information Technology    |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-48941   | Mallory Mason    | CME       | Maritime Engineering      |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-17178   | Rico Blanco      | CEng      | Electronics Engineering   |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A22-45722   | Nicole Watson    | CNAHS     | Medical Technology        |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-53516   | Neo Tagle        | CCMS      | Computer Science          |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-40161   | Bryan Johnson    | CEd       | Elementary Education      |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A22-69740   | Stephen Currie   | CAFA      | Architecture              |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-94404   | Amelia Earhart   | CIHTM     | Tourism Management        |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-37267   | Gordon Ransom    | CIHTM     | Hospitality Management    |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-15513   | Carlos Oblek     | CAFA      | Fine Arts                 |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-64046   | Ace Spades       | CCMS      | Information Technology    |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-13291   | Vincent Aguilar  | CAS       | Psychology                |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-67388   | Kyla Ren         | CBA       | Human Resource Management |      1 |
+-------------+------------------+-----------+---------------------------+--------+

Do you want to perform another operation (y/n):
```

Viewing the scholarship_applications database using view_csv().

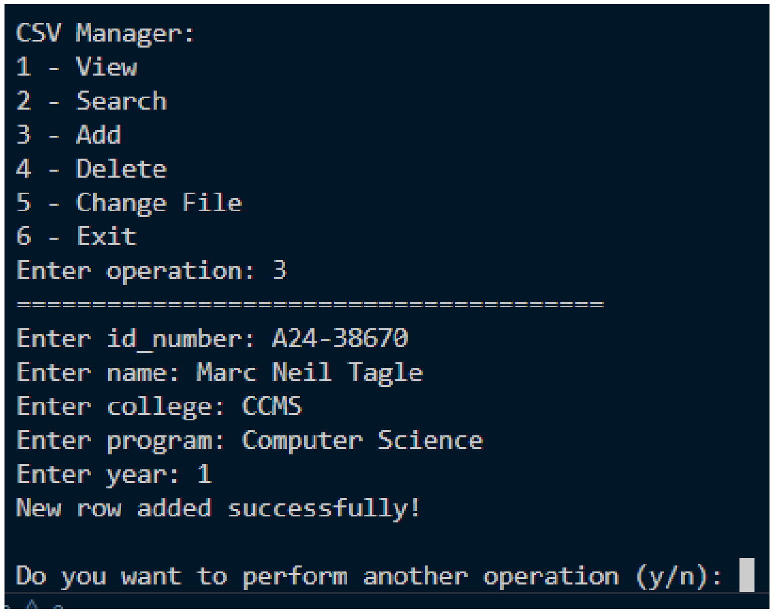
C. Searching for data



```
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 2
=======================================
Columns:
 - id_number
 - name
 - college
 - program
 - year

Enter column (or type 'exit' to cancel): college
Search: CCMS
4 row/s found:

 id_number, name, college, program, year

+-------------+-------------+-----------+-------------------------+--------+
| id_number   | name        | college   | program                 |   year |
+=============+=============+===========+=========================+========+
| A24-82295   | Nico Ariate | CCMS      | Computer Science        |      1 |
+-------------+-------------+-----------+-------------------------+--------+
| A21-87961   | Faux Vaux   | CCMS      | Information Technology  |      4 |
+-------------+-------------+-----------+-------------------------+--------+
| A24-53516   | Neo Tagle   | CCMS      | Computer Science        |      1 |
+-------------+-------------+-----------+-------------------------+--------+
| A23-64046   | Ace Spades  | CCMS      | Information Technology  |      2 |
+-------------+-------------+-----------+-------------------------+--------+

Do you want to perform another operation (y/n):
```

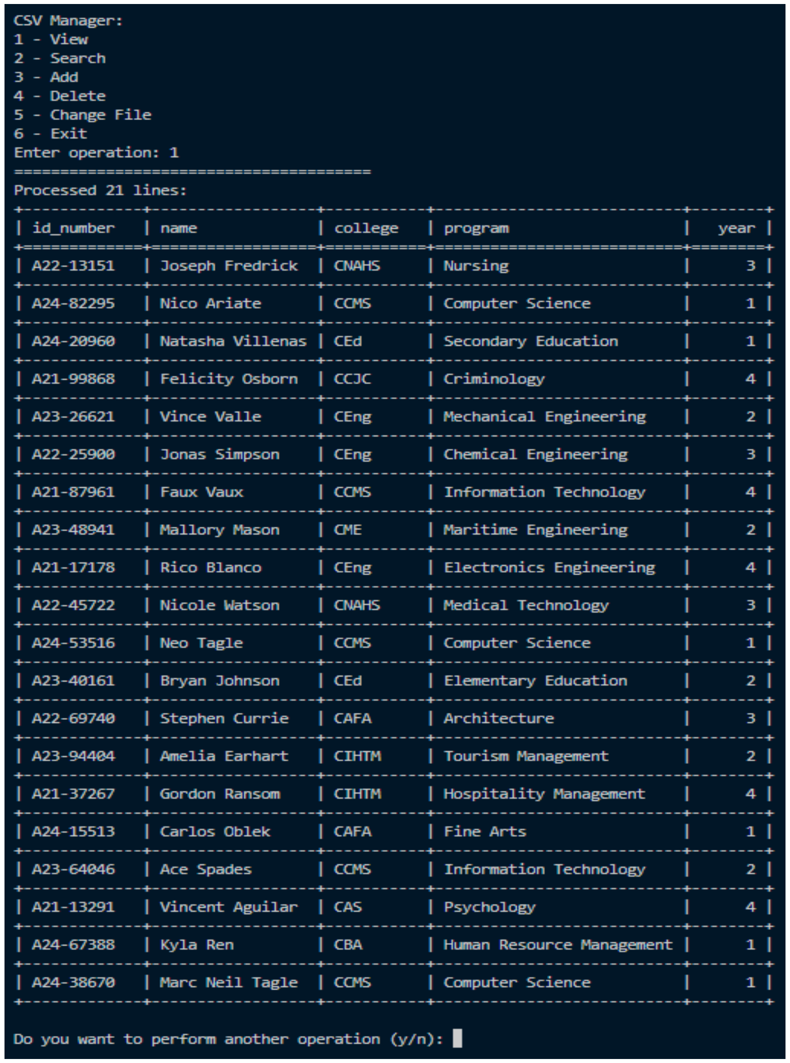
Searching rows of CCMS students.

D. Adding a new row



```
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 3
========================================
Enter id_number: A24-38670
Enter name: Marc Neil Tagle
Enter college: CCMS
Enter program: Computer Science
Enter year: 1
New row added successfully!

Do you want to perform another operation (y/n):
```

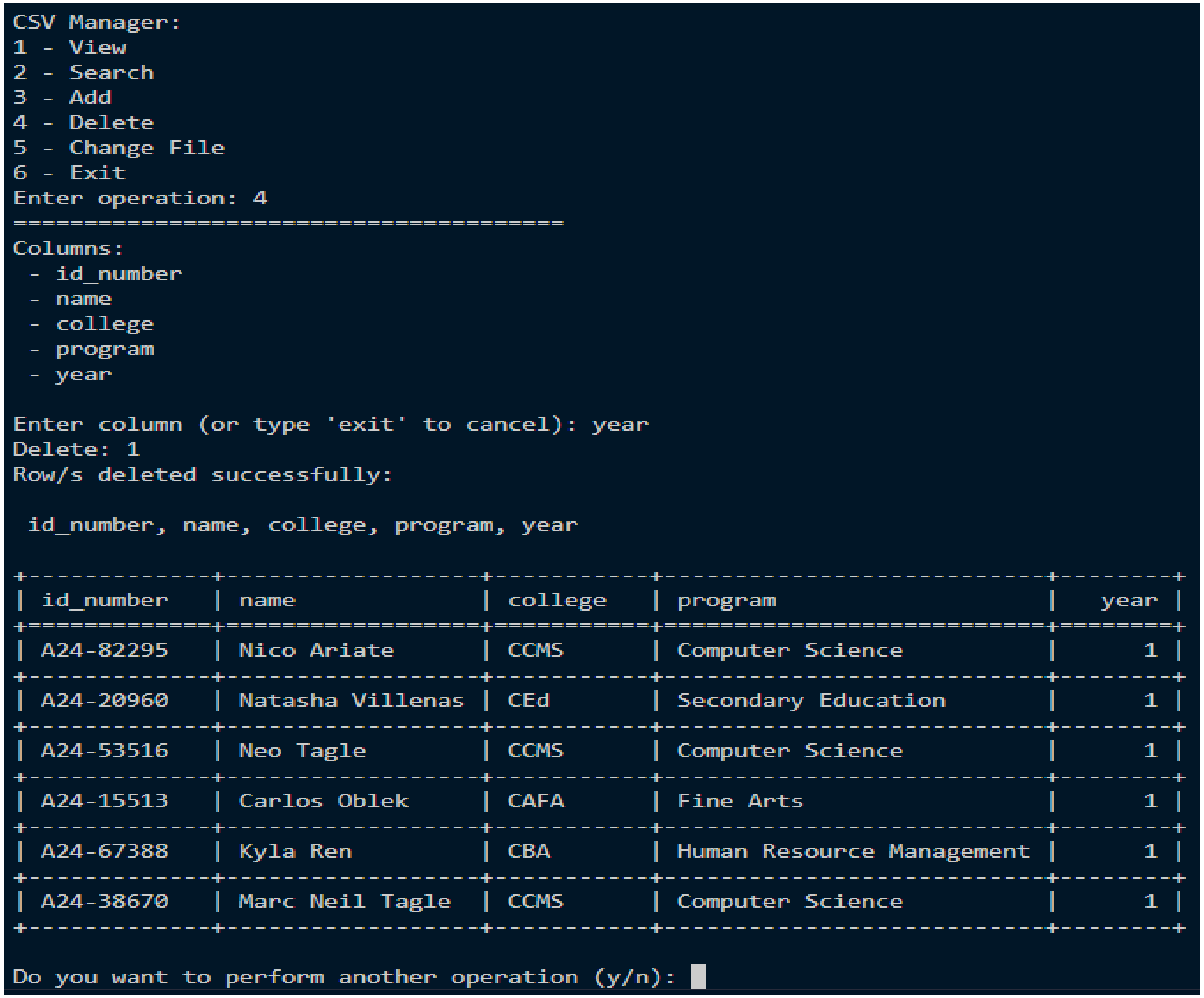
Adding a new student in the scholarship_applications database.



```
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 1
========================================
Processed 21 lines:
+-------------+------------------+-----------+---------------------------+--------+
| id_number   | name             | college   | program                   |   year |
+=============+==================+===========+===========================+========+
| A22-13151   | Joseph Fredrick  | CNAHS     | Nursing                   |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-82295   | Nico Ariate      | CCMS      | Computer Science          |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-20960   | Natasha Villenas | CEd       | Secondary Education       |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-99868   | Felicity Osborn  | CCJC      | Criminology               |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-26621   | Vince Valle      | CEng      | Mechanical Engineering    |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A22-25900   | Jonas Simpson    | CEng      | Chemical Engineering      |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-87961   | Faux Vaux        | CCMS      | Information Technology    |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-48941   | Mallory Mason    | CME       | Maritime Engineering      |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-17178   | Rico Blanco      | CEng      | Electronics Engineering   |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A22-45722   | Nicole Watson    | CNAHS     | Medical Technology        |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-53516   | Neo Tagle        | CCMS      | Computer Science          |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-40161   | Bryan Johnson    | CEd       | Elementary Education      |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A22-69740   | Stephen Currie   | CAFA      | Architecture              |      3 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-94404   | Amelia Earhart   | CIHTM     | Tourism Management        |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-37267   | Gordon Ransom    | CIHTM     | Hospitality Management    |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-15513   | Carlos Oblek     | CAFA      | Fine Arts                 |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A23-64046   | Ace Spades       | CCMS      | Information Technology    |      2 |
+-------------+------------------+-----------+---------------------------+--------+
| A21-13291   | Vincent Aguilar  | CAS       | Psychology                |      4 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-67388   | Kyla Ren         | CBA       | Human Resource Management |      1 |
+-------------+------------------+-----------+---------------------------+--------+
| A24-38670   | Marc Neil Tagle  | CCMS      | Computer Science          |      1 |
+-------------+------------------+-----------+---------------------------+--------+

Do you want to perform another operation (y/n):
```

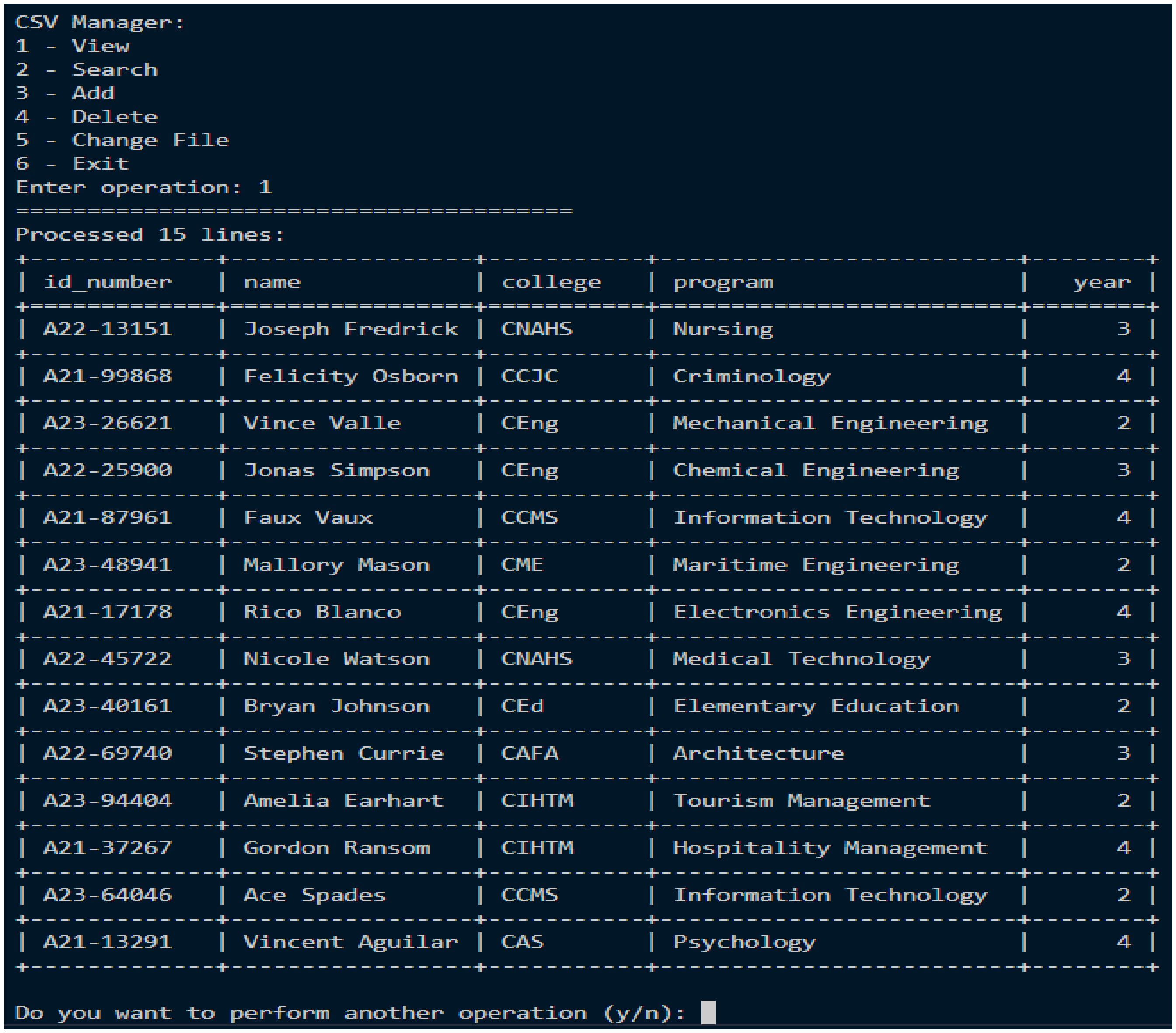
Viewing the scholarship_applications database with the added student.

E. Deleting a row



```
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 4
=======================================
Columns:
 - id_number
 - name
 - college
 - program
 - year

Enter column (or type 'exit' to cancel): year
Delete: 1
Row/s deleted successfully:

 id_number, name, college, program, year

+-------------+-------------------+-----------+----------------------------+--------+
| id_number   | name              | college   | program                    |   year |
+=============+===================+===========+============================+========+
| A24-82295   | Nico Ariate       | CCMS      | Computer Science           |      1 |
+-------------+-------------------+-----------+----------------------------+--------+
| A24-20960   | Natasha Villenas  | CEd       | Secondary Education        |      1 |
+-------------+-------------------+-----------+----------------------------+--------+
| A24-53516   | Neo Tagle         | CCMS      | Computer Science           |      1 |
+-------------+-------------------+-----------+----------------------------+--------+
| A24-15513   | Carlos Oblek      | CAFA      | Fine Arts                  |      1 |
+-------------+-------------------+-----------+----------------------------+--------+
| A24-67388   | Kyla Ren          | CBA       | Human Resource Management  |      1 |
+-------------+-------------------+-----------+----------------------------+--------+
| A24-38670   | Marc Neil Tagle   | CCMS      | Computer Science           |      1 |
+-------------+-------------------+-----------+----------------------------+--------+

Do you want to perform another operation (y/n):
```

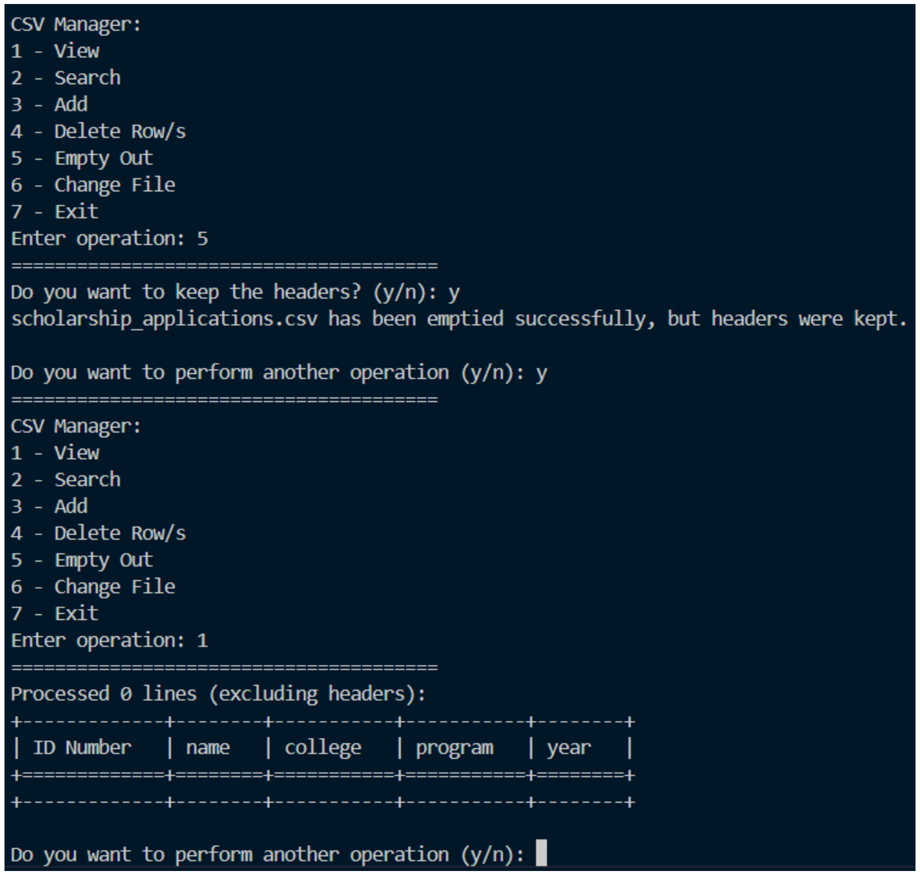
Deleting rows of 1st year applicants.



```
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 1
=======================================
Processed 15 lines:
+-------------+------------------+-----------+--------------------------+---------+
| id_number   | name             | college   | program                  |    year |
+=============+==================+===========+==========================+=========+
| A22-13151   | Joseph Fredrick  | CNAHS     | Nursing                  |       3 |
+-------------+------------------+-----------+--------------------------+---------+
| A21-99868   | Felicity Osborn  | CCJC      | Criminology              |       4 |
+-------------+------------------+-----------+--------------------------+---------+
| A23-26621   | Vince Valle      | CEng      | Mechanical Engineering   |       2 |
+-------------+------------------+-----------+--------------------------+---------+
| A22-25900   | Jonas Simpson    | CEng      | Chemical Engineering     |       3 |
+-------------+------------------+-----------+--------------------------+---------+
| A21-87961   | Faux Vaux        | CCMS      | Information Technology   |       4 |
+-------------+------------------+-----------+--------------------------+---------+
| A23-48941   | Mallory Mason    | CME       | Maritime Engineering     |       2 |
+-------------+------------------+-----------+--------------------------+---------+
| A21-17178   | Rico Blanco      | CEng      | Electronics Engineering  |       4 |
+-------------+------------------+-----------+--------------------------+---------+
| A22-45722   | Nicole Watson    | CNAHS     | Medical Technology       |       3 |
+-------------+------------------+-----------+--------------------------+---------+
| A23-40161   | Bryan Johnson    | CEd       | Elementary Education     |       2 |
+-------------+------------------+-----------+--------------------------+---------+
| A22-69740   | Stephen Currie   | CAFA      | Architecture             |       3 |
+-------------+------------------+-----------+--------------------------+---------+
| A23-94404   | Amelia Earhart   | CIHTM     | Tourism Management       |       2 |
+-------------+------------------+-----------+--------------------------+---------+
| A21-37267   | Gordon Ransom    | CIHTM     | Hospitality Management   |       4 |
+-------------+------------------+-----------+--------------------------+---------+
| A23-64046   | Ace Spades       | CCMS      | Information Technology   |       2 |
+-------------+------------------+-----------+--------------------------+---------+
| A21-13291   | Vincent Aguilar  | CAS       | Psychology               |       4 |
+-------------+------------------+-----------+--------------------------+---------+

Do you want to perform another operation (y/n):
```

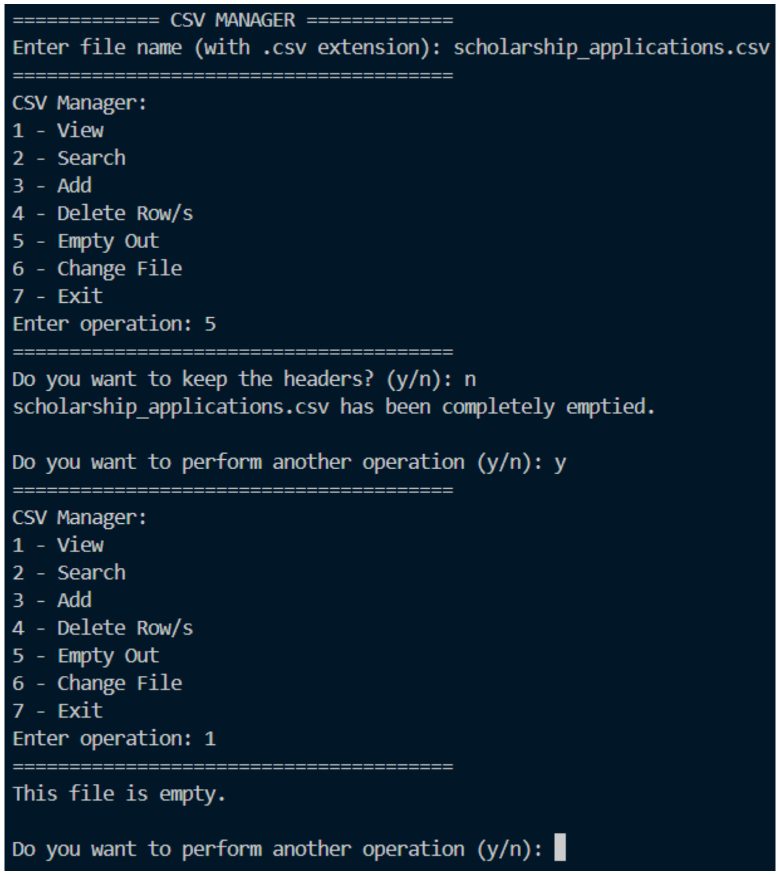
Viewing the scholarship_applications database excluding the deleted applicants.

F. Empty out file



```
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete Row/s
5 - Empty Out
6 - Change File
7 - Exit
Enter operation: 5
========================================
Do you want to keep the headers? (y/n): y
scholarship_applications.csv has been emptied successfully, but headers were kept.

Do you want to perform another operation (y/n): y
========================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete Row/s
5 - Empty Out
6 - Change File
7 - Exit
Enter operation: 1
========================================
Processed 0 lines (excluding headers):
+-------------+--------+-----------+-----------+--------+
| ID Number   | name   | college   | program   | year   |
+=============+========+===========+===========+========+
+-------------+--------+-----------+-----------+--------+

Do you want to perform another operation (y/n):
```

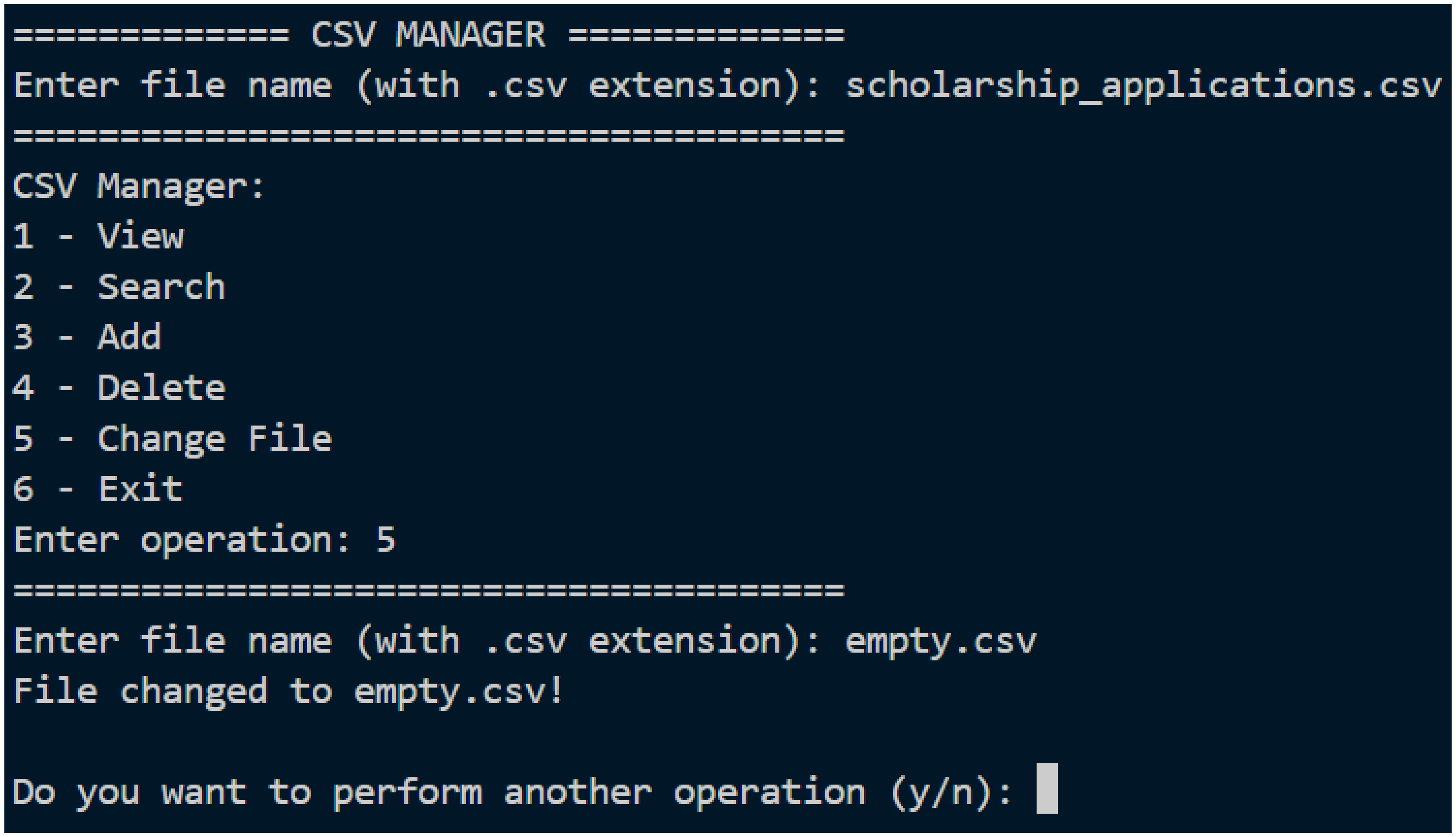
Emptying out the scholarship_application database (keeping headers).



```
============= CSV MANAGER =============
Enter file name (with .csv extension): scholarship_applications.csv
========================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete Row/s
5 - Empty Out
6 - Change File
7 - Exit
Enter operation: 5
========================================
Do you want to keep the headers? (y/n): n
scholarship_applications.csv has been completely emptied.

Do you want to perform another operation (y/n): y
========================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete Row/s
5 - Empty Out
6 - Change File
7 - Exit
Enter operation: 1
========================================
This file is empty.

Do you want to perform another operation (y/n):
```

Emptying out the scholarship_application database (not keeping headers).

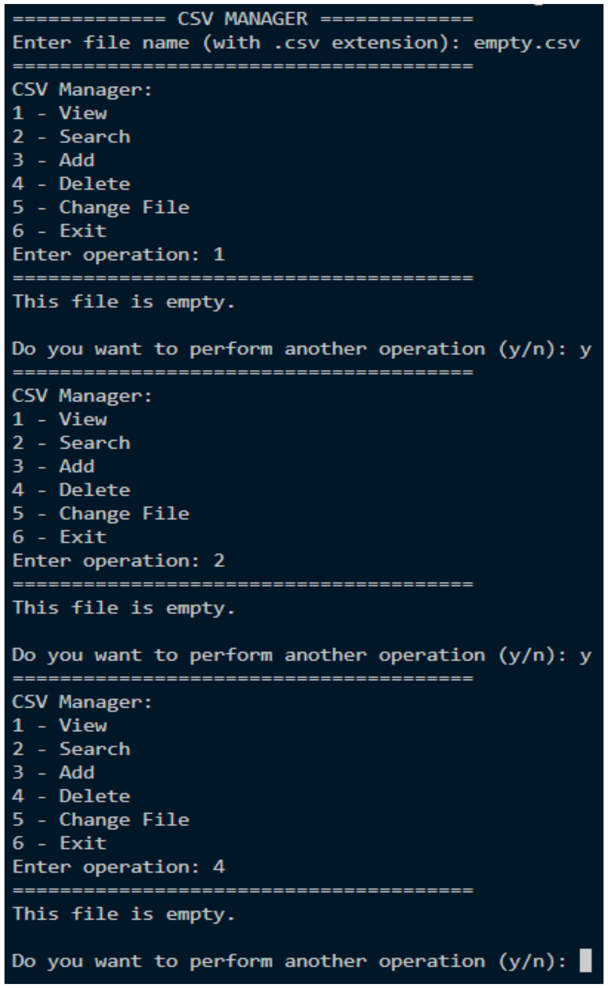
G. Changing files



```
============= CSV MANAGER =============
Enter file name (with .csv extension): scholarship_applications.csv
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 5
=======================================
Enter file name (with .csv extension): empty.csv
File changed to empty.csv!

Do you want to perform another operation (y/n): 
```

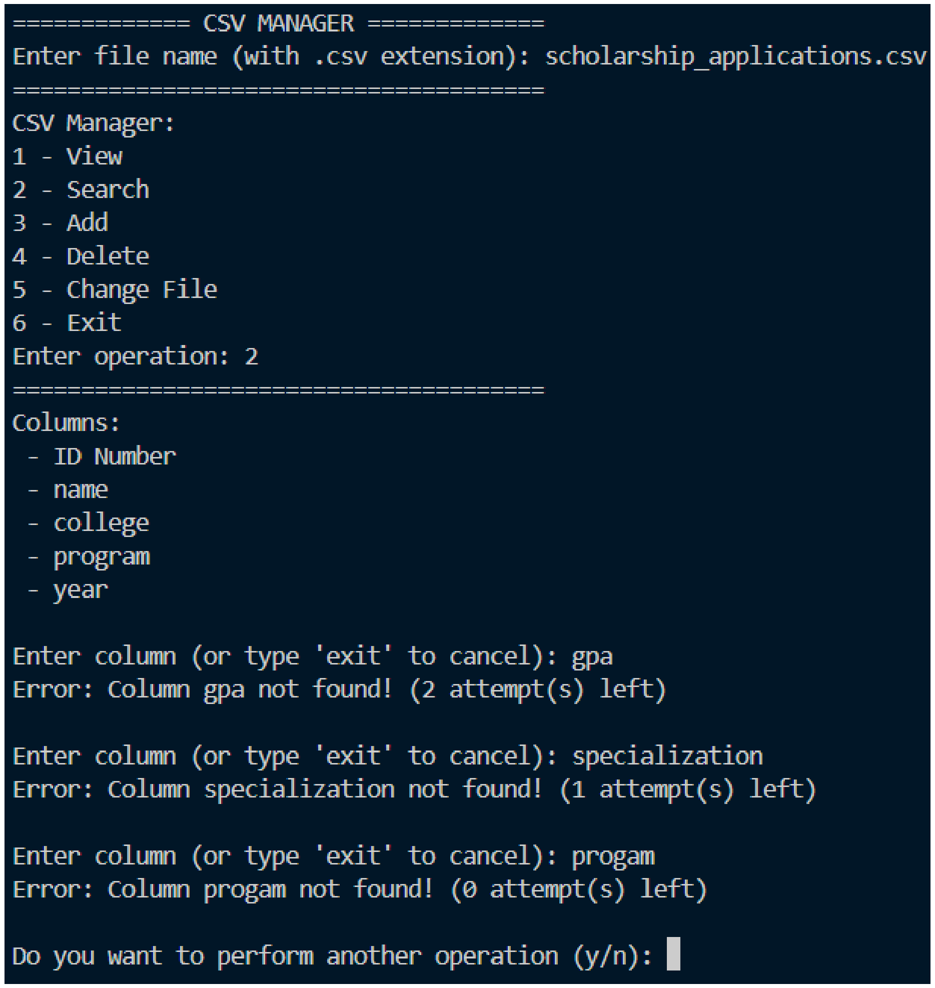
Changing CSV file to manage.

H. Error handling (e.g., invalid input, empty file.)



```
============= CSV MANAGER =============
Enter file name (with .csv extension): empty.csv
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 1
=======================================
This file is empty.

Do you want to perform another operation (y/n): y
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 2
=======================================
This file is empty.

Do you want to perform another operation (y/n): y
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 4
=======================================
This file is empty.

Do you want to perform another operation (y/n):
```

Using view_csv(), search_csv(), and delete_row() with an empty CSV file.



```
============= CSV MANAGER =============
Enter file name (with .csv extension): scholarship_applications.csv
=======================================
CSV Manager:
1 - View
2 - Search
3 - Add
4 - Delete
5 - Change File
6 - Exit
Enter operation: 2
=======================================
Columns:
 - ID Number
 - name
 - college
 - program
 - year

Enter column (or type 'exit' to cancel): gpa
Error: Column gpa not found! (2 attempt(s) left)

Enter column (or type 'exit' to cancel): specialization
Error: Column specialization not found! (1 attempt(s) left)

Enter column (or type 'exit' to cancel): progam
Error: Column progam not found! (0 attempt(s) left)

Do you want to perform another operation (y/n):
```

Entering invalid columns in search(). This also works for delete().