**Student Grades Management System**

**Name:** Bryan Yadiel Caban Rodriguez

**Date Created:** March 23, 2025

**Program Description:** A Student Grades Management System that allows instructors to create and view student exam grades. The system consists of two main programs: one to create a CSV file containing student information and exam grades, and another to read and display the grades in a tabular format.

**Functions used in the Program:**

1. **Function Name:** create\_grades\_file()

**Description:** Create a CSV file with student information and exam grades.

**Parameters:** None

**Variables:**

* + num\_students (int) - stores the number of students to enter
  + student\_data (dict) - stores information for each student

**Logical Steps:**

* + Prompt user for the number of students
  + Open grades.csv in write mode
  + Create a CSV DictWriter with appropriate field names
  + Write the header row
  + For each student:
    - Collect name and exam grades
    - Write the data as a row in the CSV file

**Returns:** None

1. **Function Name:** display\_grades\_file()

**Description:** Read the grades.csv file and display the data in a formatted table.

**Parameters:** None

**Variables:**

* + row (dict) - stores each row of data from the CSV file

**Logical Steps:**

* + Open grades.csv in read mode
  + Create a CSV DictReader
  + Print a formatted header
  + For each row in the file:
    - Format and print the data

**Returns:** None

1. **Function Name:** main()

**Description:** Provide a user interface for the system.

**Parameters:** None

**Variables:**

* + choice (str) - stores the user's menu selection

**Logical Steps:**

* + Display a menu with options
  + Based on user selection:
    - Create a new grades file
    - Display an existing grades file
    - Exit the program

**Returns:** None

**Overall Program Logical Flow:**

1. Program starts and displays the main menu
2. Based on user selection:
   * If create: collect student data and write to CSV file
   * If display: read and show the data from the existing CSV file
   * If exit: end the program
3. Return to the main menu after completing an operation

**Data Structure:**

The data is stored in a CSV file with the following structure:

| **First Name** | **Last Name** | **Exam 1** | **Exam 2** | **Exam 3** |
| --- | --- | --- | --- | --- |
| John | Doe | 85 | 92 | 78 |
| Jane | Smith | 90 | 88 | 95 |

**Implementation Details:**

* **Libraries Used:**
  + **csv**: For reading and writing CSV files
  + **built-in Python functions**: For input/output operations
* **Error Handling:**
  + Validates that the number of students and exam grades are integers
  + Checks if the grades.csv file exists
  + Captures and reports any exceptions that occur during file operations

**Testing Plan:**

1. Test create\_grades\_file() with various numbers of students
2. Test input validation for non-integer values
3. Test display\_grades\_file() with different CSV file formats
4. Test the full workflow from creating a file to displaying it
5. Test error handling when the file doesn't exist

**Limitations and Future Improvements:**

* The system currently does not calculate statistics (averages, highest/lowest scores)
* Data validation could be enhanced (e.g., grade ranges)
* A more sophisticated UI could be implemented
* The system could be extended to support file editing and record deletion

**Link to your repository:** https://github.com/xXTeinsXx/COP2373