**Technical Design Document Template**

**Name:** Alanys Suazo

**Date Created: 10/26/25**

**Program Description:**

This program is designed to organize student scores and information into a CSV file and show the results to the user. It takes user input and writes the info into the file, then it reads the file with all the changes made and shows the user a table with the organized information.

**Functions used in the Program (list in order as they are called):**

1. **Function Name:** reader program

**Description:** This function starts the program. It prompts the user for the number of students, calls the writer\_program to collect and store student data, and then reads and displays the contents of the CSV file in a formatted table.

**Parameters:** none

**Variables:**

* loop\_interval: stores the number of students entered by the user
* file: represents the opened CSV file in read mode
* csvreader: used to iterate through rows in the CSV file
* row: represents each row of student data read from the CSV file

**Logical Steps:**

1. Display a welcome message
2. Prompt the user to enter the number of students
3. Convert the input to an integer and store it in loop\_interval
4. Call writer\_program(loop\_interval) to collect and save student data
5. Print a formatted header for the output table
6. Open grades.csv in read mode
7. Create a CSV reader object
8. Skip the header row using the next(csvreader) command
9. Loop through each row and print it using formatted column alignment

**Returns:** nothing

2. **Function Name:** writer\_program

**Description:** This function collects student names and exam scores, validates the input, and writes the data to a CSV file named grades.csv.

**Parameters:**

* loop\_interval: This helps the program to know how many times to run the loop

**Variables:**

* file (file object): represents the opened CSV file in write mode
* writer (csv.writer object): used to write rows to the CSV file
* i (int): loop counter for student entries
* first (str): first name of the student
* last (str): last name of the student
* exam1, exam2, exam3 (float): scores for each of the three exams

**Logical Steps:**

1. Open grades.csv in write mode, overwriting any existing content
2. Create a CSV writer object
3. Write the header row: First Name, Last Name, Exam 1, Exam 2, Exam 3
4. Loop loop\_interval times to collect data for each student
5. For each student:

* Prompt for first name and validate using the .isalpha() inside a while loop
* Prompt for last name and validate similarly
* Prompt for each exam score and validate using try/except and range checks (0–100)
* Write the validated data as a row in the CSV file

**Returns:** nothing because it writes the information to the csv file.

**Logical Steps:**

1. reader\_program function is called first when the program runs
2. reader\_program prompts the user for the number of students
3. reader\_program calls writer\_program with that number
4. writer\_program collects and writes student data to the CSV file
5. reader\_program then reads and prints the contents of the CSV file in table format

**Link to your repository:** [Alanys-SG/COP2373 at master](https://github.com/Alanys-SG/COP2373/tree/master)

**Output Screenshot: (make sure big enough so I can see)**

**A screenshot of a computer program

AI-generated content may be incorrect.**