Python and Bash Assignment

Name: Kundan Agrawal

# 1. Grade Checker

def grade\_checker():

    """

    Takes a score as input from the user and prints the grade as per the criteria:

    90+ : 'A', 80-89 : 'B', 70-79 : 'C', 60-69 : 'D', Below 60 : 'F'

    """

    try:

        score = float(input("Enter the score: "))

        if score >= 90:

            grade = 'A'

        elif score >= 80:

            grade = 'B'

        elif score >= 70:

            grade = 'C'

        elif score >= 60:

            grade = 'D'

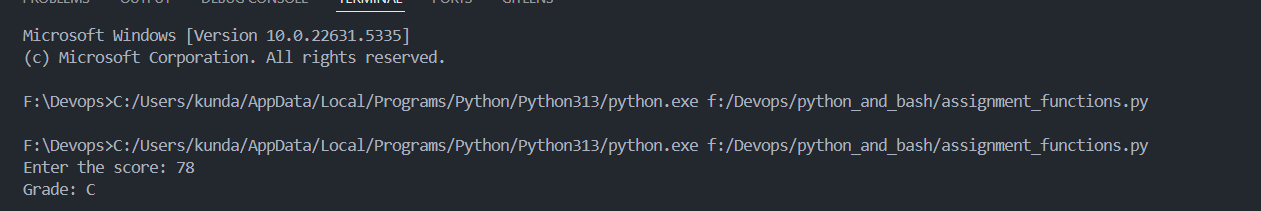
        else:

            grade = 'F'

        print(f"Grade: {grade}")

    except ValueError:

        print("Invalid input. Please enter a numeric value.")

Screenshot: 

# 2. Student Grades

def manage\_student\_grades():

    """

    Allows user to add, update, and print student grades using a dictionary.

    """

    student\_grades = {}

    while True:

        print("\nOptions: 1-Add 2-Update 3-Print 4-Exit")

        choice = input("Enter your choice: ")

        if choice == '1':

            name = input("Enter student name: ")

            grade = input("Enter grade: ")

            if name in student\_grades:

                print(f"{name} already exists. Use update option.")

            else:

                student\_grades[name] = grade

                print(f"Added {name} with grade {grade}.")

        elif choice == '2':

            name = input("Enter student name to update: ")

            if name in student\_grades:

                grade = input("Enter new grade: ")

                student\_grades[name] = grade

                print(f"Updated {name} to grade {grade}.")

            else:

                print(f"{name} not found.")

        elif choice == '3':

            print("\nAll Student Grades:")

            for name, grade in student\_grades.items():

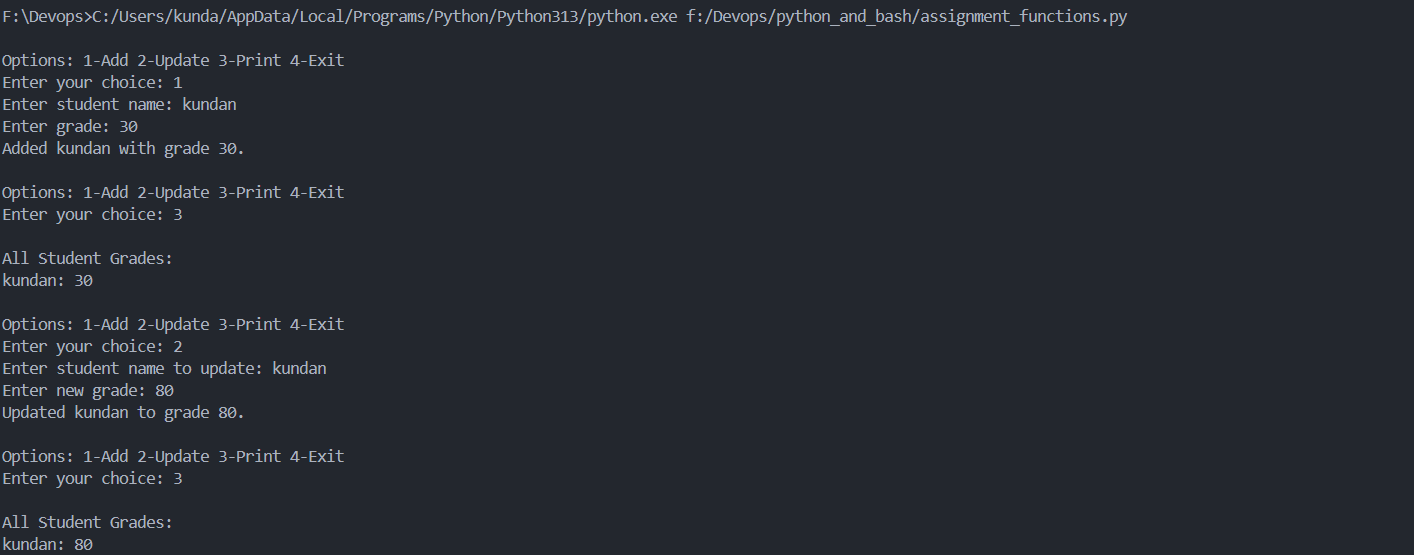
                print(f"{name}: {grade}")

        elif choice == '4':

            break

        else:

            print("Invalid choice. Try again.")

Screenshot:   


# 3. Write to a file

def write\_to\_file(filename, content):

    """

    Writes the given content to a text file.

    """

    with open(filename, 'w') as f:

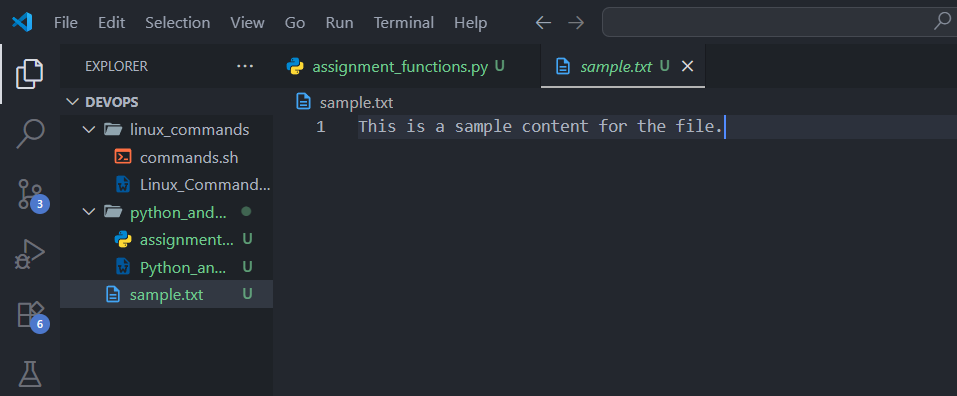
        f.write(content)

    print(f"Content written to {filename}.")

content = "This is a sample content for the file."

filename = "sample.txt"

write\_to\_file(filename, content)

Screenshot:   


# 4. Read from a file

filename = "sample.txt"

def read\_from\_file(filename):

    """

    Reads and prints the content of a text file.

    """

    try:

        with open(filename, 'r') as f:

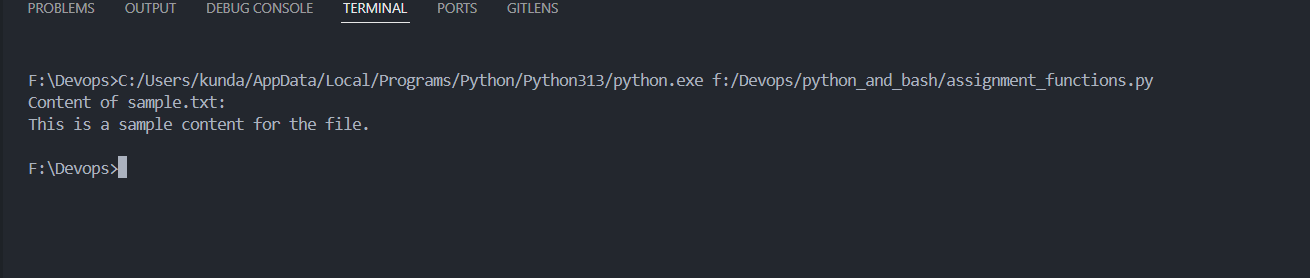
            data = f.read()

        print(f"Content of {filename}:\n{data}")

    except FileNotFoundError:

        print(f"File {filename} not found.")

read\_from\_file(filename)

Screenshot:   


# GitHub Repository

https://github.com/Kundan1804/Devops/tree/main/python\_and\_bash