1. Grade Checker

Take a score as input and print the grade based on the following:

90+ : "A"

80-89 : "B"

70-79 : "C"

60-69 : "D"

Below 60 : "F"

here we used a basic if else statement to carry out marks and all.

Answer: -

'''

Take a score as input and print the grade based on the following:

90+ : "A"

80-89 : "B"

70-79 : "C"

60-69 : "D"

Below 60 : "F"

here we used a basic if else statement to carry out marks and all.

'''

score = int(input("Enter your score: "))

if score >= 90:

    print("A")

elif score >= 80:

    print("B")

elif score >= 70:

    print("C")

elif score >= 60:

    print("D")

else:

    print("F")

2 Student Grades

Create a dictionary where the keys are student names and the values are their grades. Allow the user to:

Add a new student and grade.

Update an existing student’s grade.

Print all student grades.

Used dictionary and basic operations. Using if else

Answer: -

'''

Create a dictionary where the keys are student names and the values are their grades. Allow the user to:

Add a new student and grade.

Update an existing student’s grade.

Print all student grades.

Used dictionary and basic operations. Using if else:

'''

students = {}

def add\_student(name, grade):

    students[name] = grade

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

def update\_student(name, grade):

    if name in students:

        students[name] = grade

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

    else:

        print(f"{name} not found in the records.")

def print\_grades():

    if students:

        print("Student Grades:")

        for name, grade in students.items():

            print(f"{name}: {grade}")

    else:

        print("No students found.")

while True:

    print("\nOptions:")

    print("1. Add student")

    print("2. Update student grade")

    print("3. Print all student grades")

    print("4. Exit")

    choice = input("Enter your choice: ")

    if choice == '1':

        name = input("Enter student's name: ")

        grade = input("Enter student's grade: ")

        add\_student(name, grade)

    elif choice == '2':

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

        grade = input("Enter new grade: ")

        update\_student(name, grade)

    elif choice == '3':

        print\_grades()

    elif choice == '4':

        break

    else:

        print("Invalid choice, please try again.")

# This code allows users to manage a list of students and their grades using a dictionary.

3.Write to a File

Write a program to create a text file and write some content to it.

Using file functions like write and open.

Answer: -

def create\_and\_write\_file(filename, content):

    try:

        with open(filename, 'w') as file:

            file.write(content)

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

    except Exception as e:

        print(f"An error occurred: {e}")

# Example usage

filename = 'example.txt'

content = "This is some example content."

create\_and\_write\_file(filename, content)

4. Read from a File

We used open in read mode and file.read to read and print to display.

Answer: -

# filepath: d:\ALL APPS\Desktop\2025-2026\devops\python\_assignment\4\_program.py

"""

4. Read from a File

We use open in read mode and file.read to read and print to display.

"""

def read\_file(filename):

    try:

        # Open the file in read mode ('r')

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

            # Read the entire content of the file

            content = file.read()

            # Display the content

            print(f"Content of {filename}:")

            print(content)

    except FileNotFoundError:

        print(f"Error: The file '{filename}' was not found.")

    except Exception as e:

        print(f"An error occurred: {e}")

# Example usage

if \_\_name\_\_ == "\_\_main\_\_":

    # Using the example.txt file that was created by #\_question.py

    filename = "example.txt"

    read\_file(filename)