1. **Grade Checker**

**Code**:

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

if score >= 90:

print("Grade: A")

elif score >= 80:

print("Grade: B")

elif score >= 70:

print("Grade: C")

elif score >= 60:

print("Grade: D")

else:

print("Grade: F")

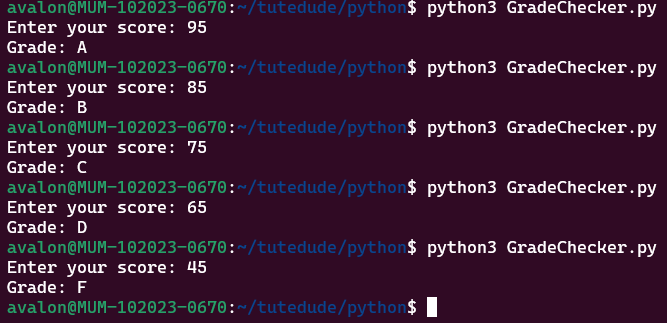
**Explanation**:

* This program takes an integer input from the user (the score).
* Using if-elif-else statements, it checks the range in which the score falls.
* Based on the score, it prints the corresponding grade.
* We used conditional statements (if, elif, else) to make decisions.

**ScreenShot**:

A screenshot of a computer

AI-generated content may be incorrect.



1. **Student Grades using Dictionary**

**Code:**

**student\_grades = {}**

**while True:**

**print("\n1. Add new 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 name: ")**

**grade = input("Enter student grade: ")**

**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("Student not found.")**

**elif choice == "3":**

**print("\nStudent Grades:")**

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

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

**elif choice == "4":**

**print("Exiting program.")**

**break**

**else:**

**print("Invalid choice. Try again.")**

**Explanation:**

* We created a dictionary to store student names and their grades.
* The user is shown a menu to choose from (add, update, print, exit).
* Based on the input, we perform different actions using if-elif-else statements.
* We used .items() to loop through dictionary key-value pairs.
* The while True loop continues until the user chooses to exit.

**ScreenShot:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Write to a File**

**Code:**

file = open("output.txt", "w")

file.write("Hello! This is a sample text written to a file.\n")

file.write("This is the second line.\n")

file.close()

print("File written successfully.")

**Explanation:**

* We used the open() function in "w" (write) mode to create and write to a file named output.txt.
* The write() function is used to write text to the file.
* The close() function is used to properly close the file after writing.

**ScreenShot:**

A screenshot of a computer

AI-generated content may be incorrect.

A computer screen with white text

AI-generated content may be incorrect.

1. **Read from a File**

**Code:**

file = open("output.txt", "r")

content = file.read()

print("File contents:")

print(content)

file.close()

**Explanation:**

* We opened the file using open() in "r" (read) mode.
* The read() function reads all the content from the file into a string.
* We printed the contents on the screen.
* Finally, the file is closed using close() to free up resources.

**ScreenShot:**

**A computer screen shot of a black screen

AI-generated content may be incorrect.A computer screen shot of a computer code

AI-generated content may be incorrect.**