**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.

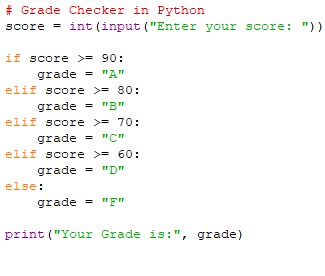
**Code:**

# Grade Checker in Python score = int(input("Enter your 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("Your Grade is:", grade)

**Screenshot :**



**Explanation:**

* The program asks the user to enter their score.
* It checks the score using if-elif-else conditions.
* Each range (90+, 80–89, 70–79, 60–69, below 60) is assigned a grade.
* The program then prints the correct grade.

**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:

**Code:**

# Student Grades using Dictionary students = {}

while True:

print("\n1. Add Student") print("2. Update Grade") print("3. Print All") print("4. Exit")

choice = int(input("Enter choice: "))

if choice == 1:

name = input("Enter student name: ") grade = input("Enter grade: ") students[name] = grade elif choice == 2:

name = input("Enter student name to update: ") if name in students:

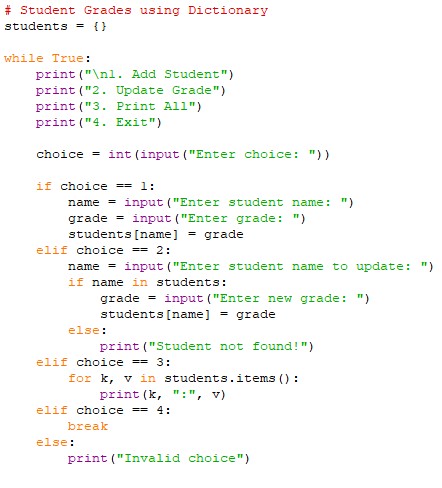
grade = input("Enter new grade: ") students[name] = grade else:

print("Student not found!") elif choice == 3: for k, v in students.items(): print(k, ":", v) elif choice == 4:

break else:

print("Invalid choice")

**Screenshot:**



**Explanation:**

* A **dictionary** students is used to store student names (keys) and grades (values).
* The program shows a menu with four options: **add, update, print all, exit**.
* If the user adds a student, the name and grade are stored in the dictionary.
* If updating, it checks if the student exists before changing the grade.
* Printing displays all student names with their grades.

1. 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.

**Code:**

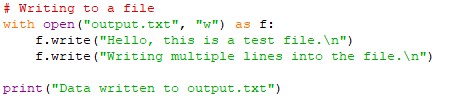
# Writing to a file

with open("output.txt", "w") as f:

f.write("Hello, this is a test file.\n")

f.write("Writing multiple lines into the file.\n")

print("Data written to output.txt") **Screenshot:**



**Explanation:**

* + open("output.txt", "w") opens a file in **write mode** (creates it if not present).
  + f.write() is used to put text into the file.
  + with ensures the file closes automatically after writing.
  + At the end, the program shows a message that writing is done.

1. Read from a File

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

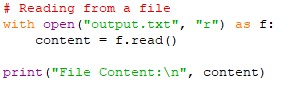
**Code:**

# Reading from a file with open("output.txt", "r") as f:

content = f.read()

print("File Content:\n", content)

**Screenshot:**



**Explanation:**

* + The program opens output.txt in **read mode**.
  + f.read() reads the complete content of the file.
  + The content is stored in the variable content.
  + Finally, it prints the file content to the screen.

**Github repo link**: https://github.com/chaudhary2001/python-assignment.git