Python File Handling & Student Grades Assignment

Question 1: Grade Checker

Task: Take a score as input and print the grade.

- **Code:**

Score = int(input(“Enter your score:”))

if score >= 90:

print(“Grade: A”)

elif score >= 80:

print(“Grade: B”)

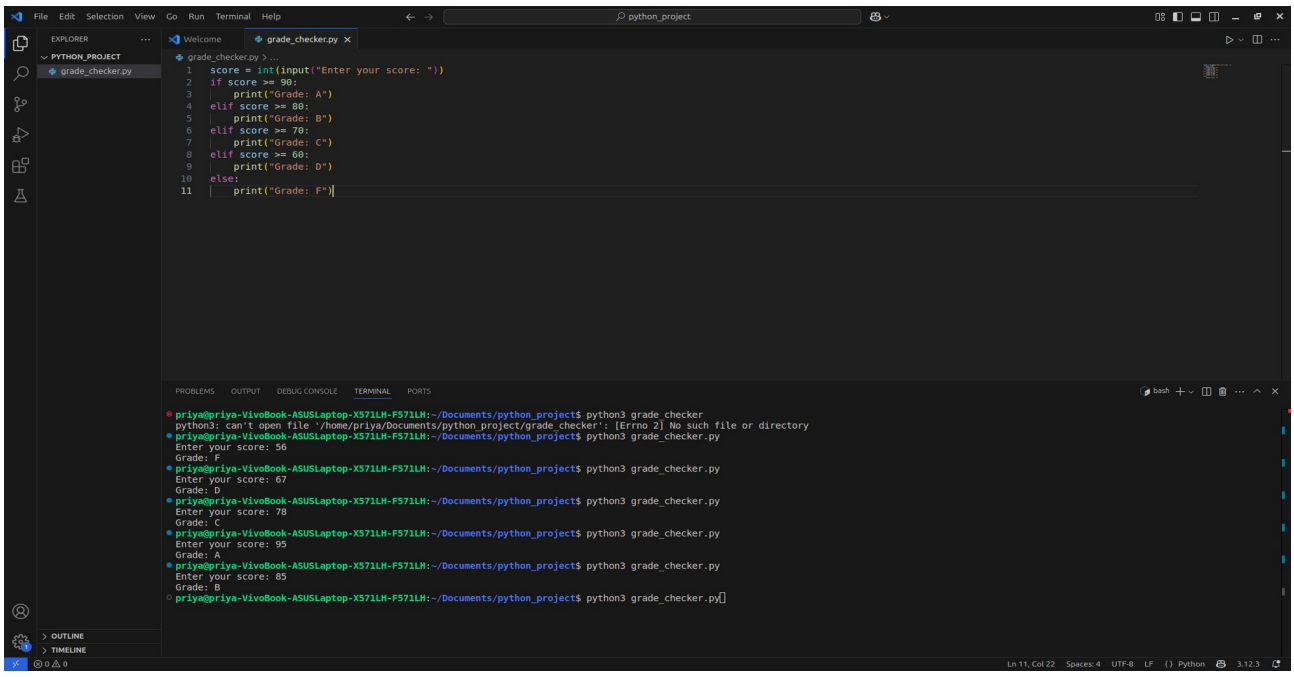
elif score >= 70:

print(“Grade: C”)

elif score >= 60:

print(“Grade: D”)

else:
print(“Grade: F”)



Explanation :

Checks the input score using `if-elif-else` statements and prints a corresponding grade.

Question 2: Student Grades Using Dictionary

Task: Add, update, and display student grades.

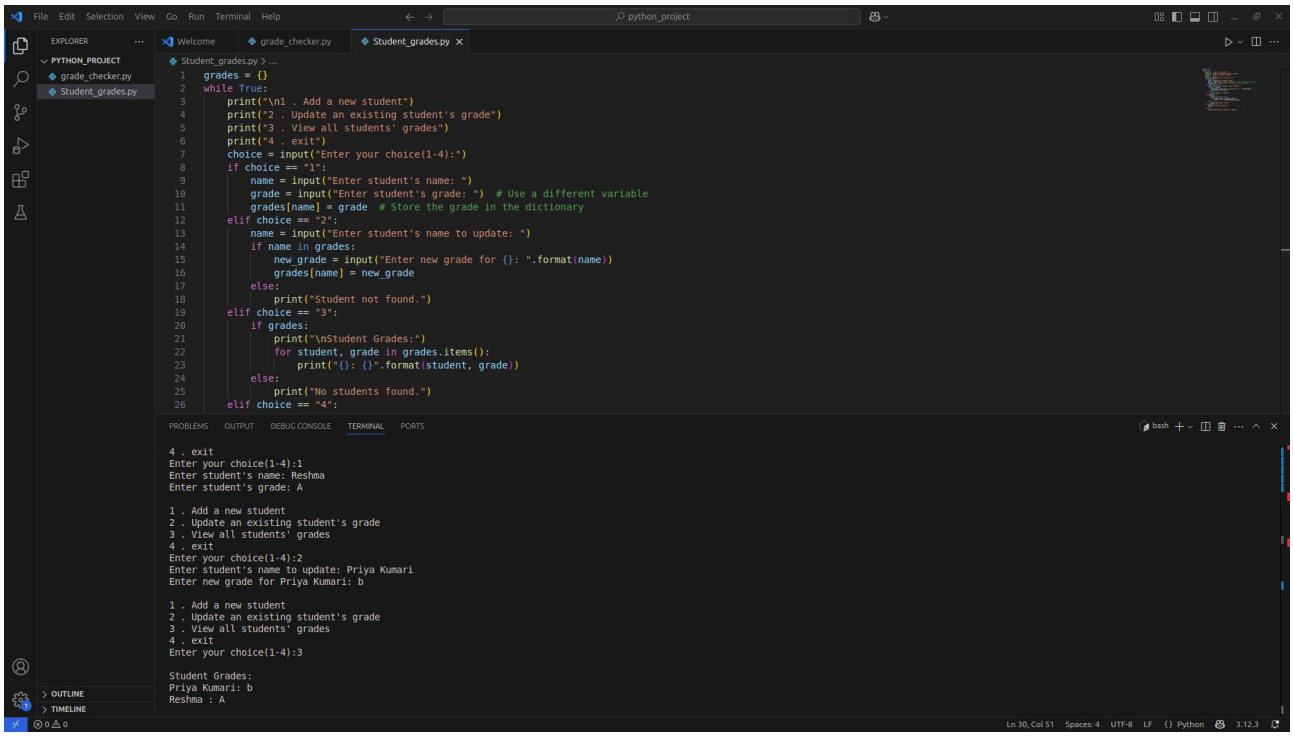
- ## Code:

```
grades = {}
while True:
print("\n1 . Add a new student")
print("2 . Update an existing student's grade")
print("3 . View all students' grades")
print("4 . exit")
choice = input("Enter your choice(1-4):")
if choice == "1":
name = input("Enter student's name: ")
grade = input("Enter student's grade: ") # Use a different variable
grades[name] = grade # Store the grade in the dictionary
elif choice == "2":
name = input("Enter student's name to update: ")
if name in grades:
new_grade = input("Enter new grade for {}: ".format(name))
grades[name] = new_grade
else:
print("Student not found.")
elif choice == "3":
if grades:
print("\nStudent Grades:")
for student, grade in grades.items():
print("{}: {}".format(student, grade))
else:
print("No students found.")
elif choice == "4":
print("Exiting the program.")
```

break

else:

print("Invalid choice. Please try again.")



Explanation :
Uses a dictionary `grades` to store student names and their grades.

- Provides a menu to add, update, or view grades.
- Uses a loop to keep the menu running until "Exit" is chosen.

Question 3: Write to a File

Task: Create a file and write content into it.

- **Code:**

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

f.write("This is my first file written using Python!\n")

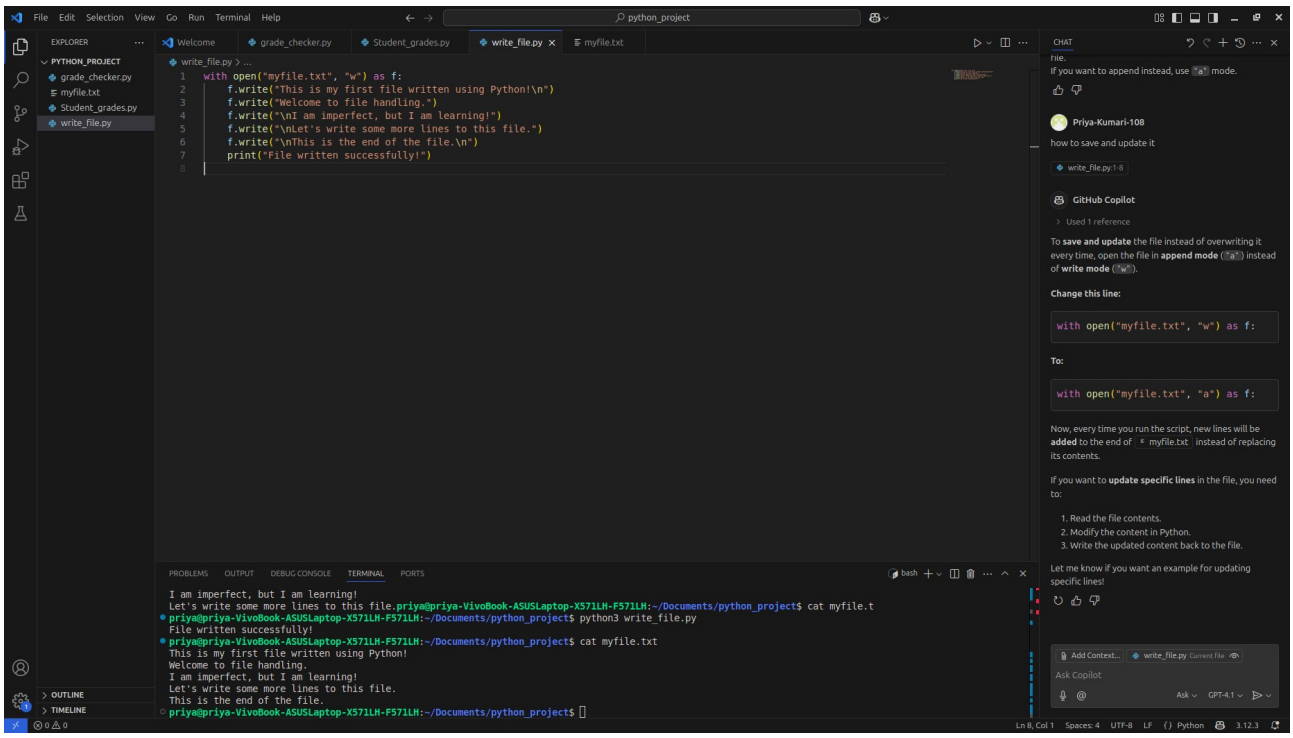
f.write("Welcome to file handling.")

f.write("\nI am imperfect, but I am learning!")

f.write("\nLet's write some more lines to this file.")

f.write("\nThis is the end of the file.\n")

print("File written successfully!")



Explanation :

`open("myfile.txt", "w")`: Opens or creates a file named myfile`.txt` in **write mode**.

- `file.write(...)`: Writes content to the file.
- `with` automatically closes the file when done.

Question 4: Read from the File

Task: Open the file created in Q3 and read its contents.

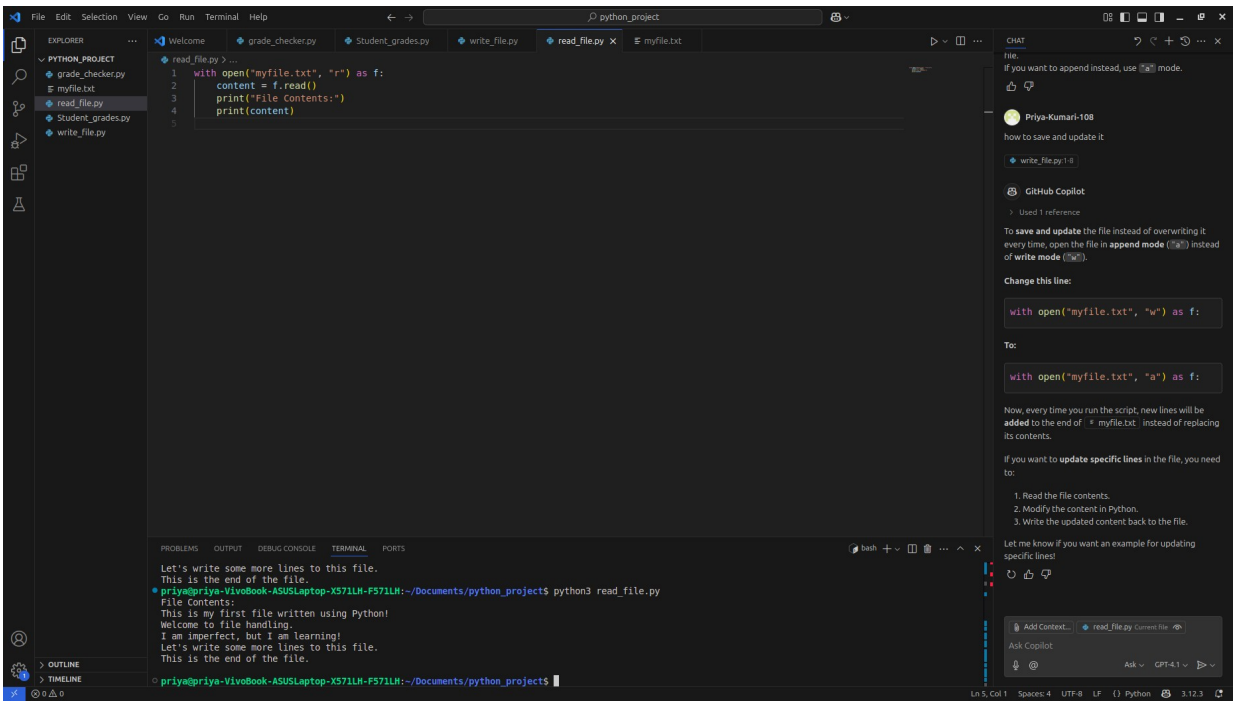
- **Code:**

with open("myfile.txt", "r") as f:

content = f.read()

print("File Contents:")

print(content)



Explanation :

- `open("myfile.txt", "r")`: Opens the same file in **read mode**.
- `file.read()`: Reads the entire content of the file.
- `print(content)`: Displays the content in the terminal.