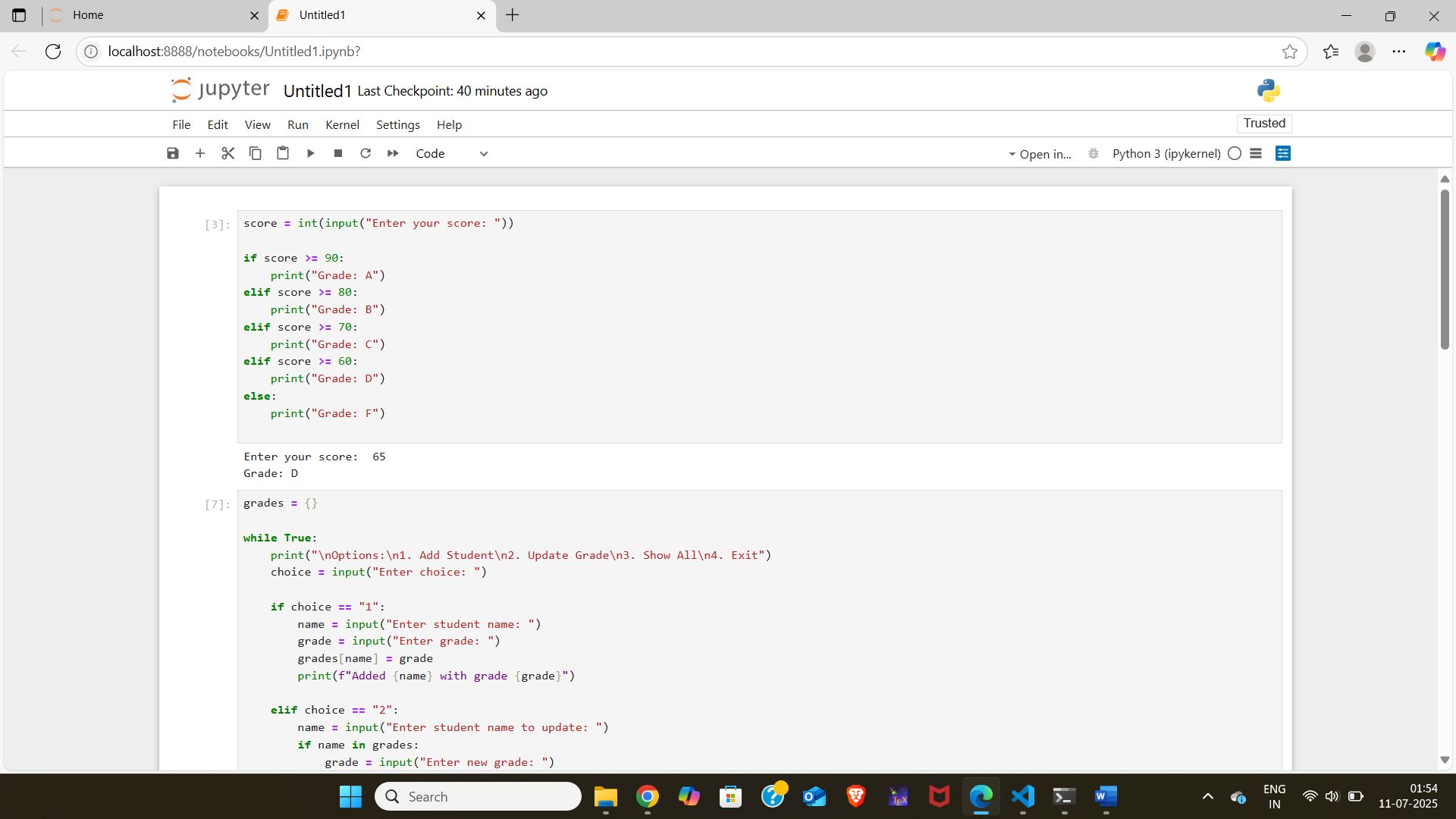
# Python Assignment: Basic Programs

## 1. Grade Checker

This program takes a numeric score as input and prints the corresponding grade.  
It uses if-elif-else conditions to determine the grade category:  
  
- 90 and above → Grade A  
- 80–89 → Grade B  
- 70–79 → Grade C  
- 60–69 → Grade D  
- Below 60 → Grade F

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")

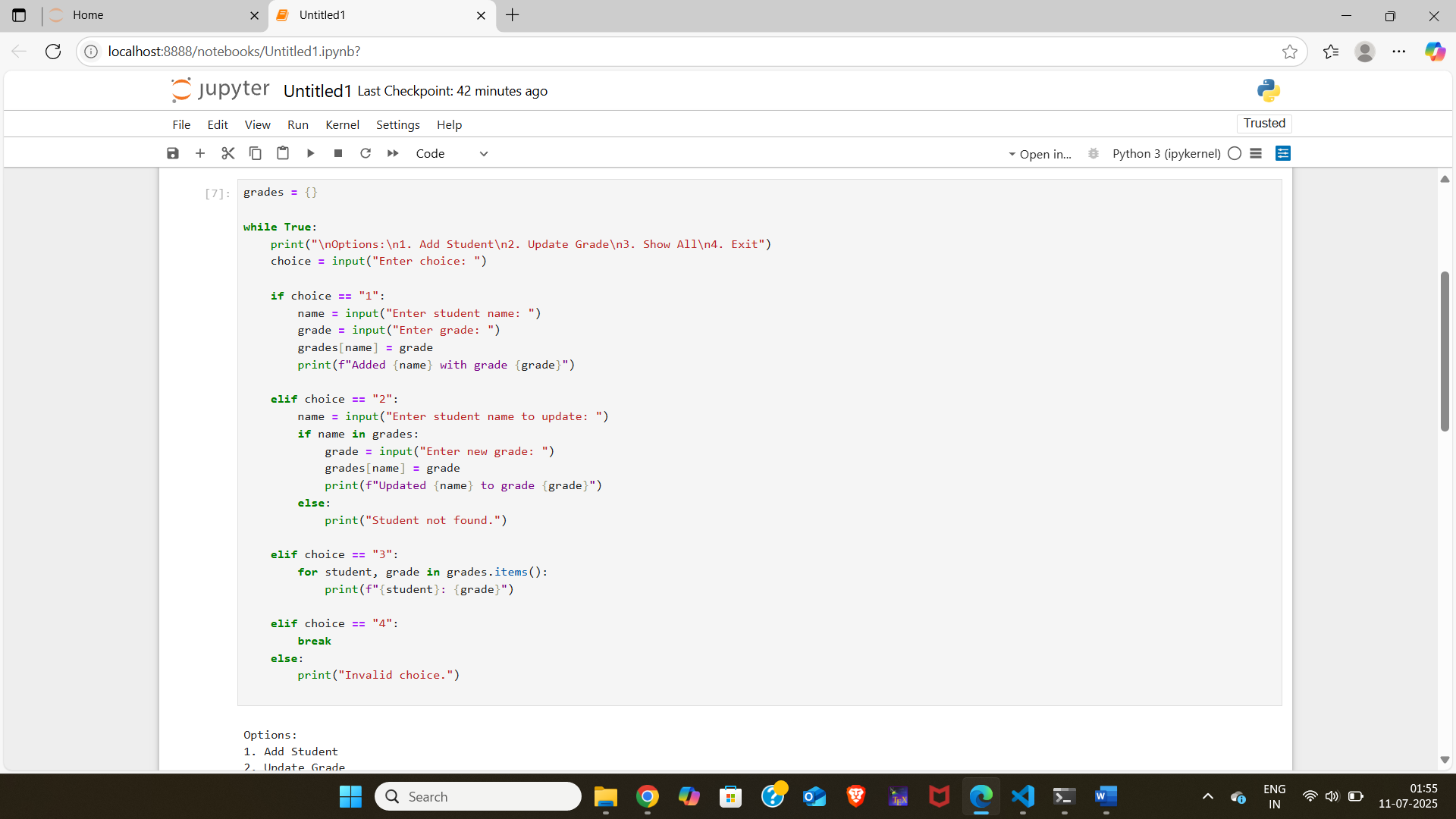
Sample Output:  
Enter your score: 92  
Grade: A  


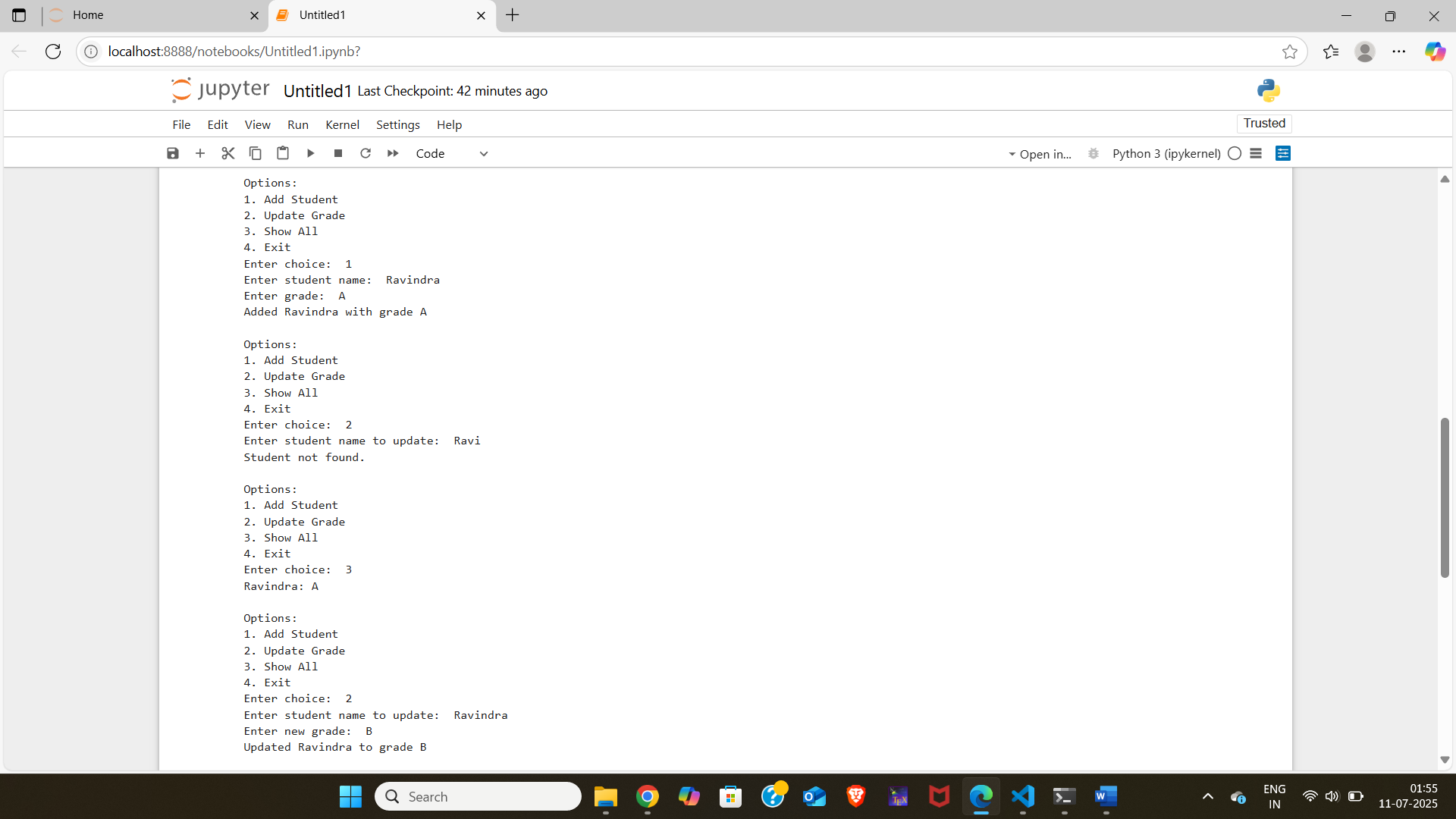
## 2. Student Grades Dictionary

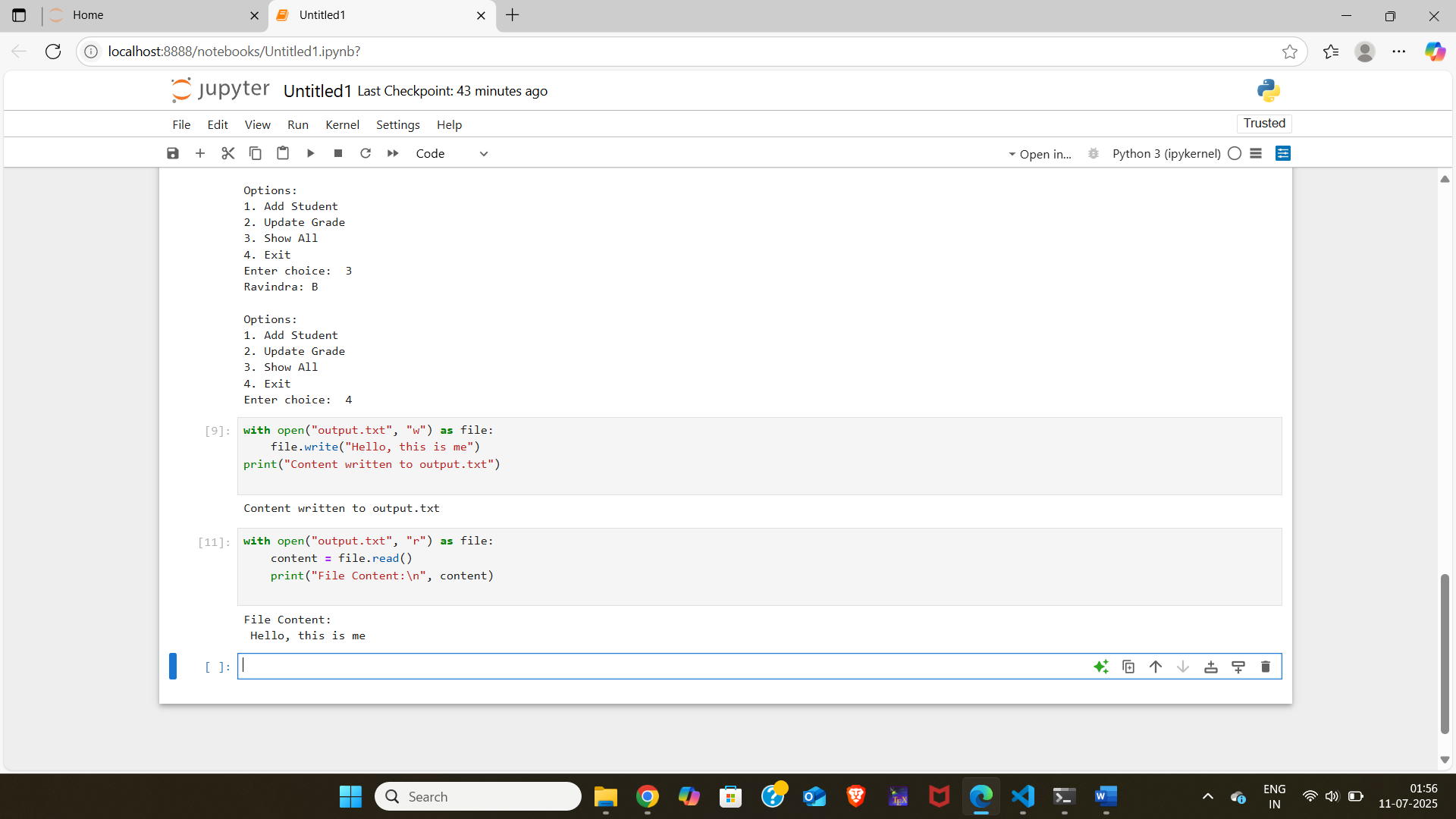
This program manages student grades using a Python dictionary. It allows users to:  
1. Add a new student and grade.  
2. Update an existing student’s grade.  
3. Display all student grades.  
4. Exit the program.

Code:

grades = {}  
  
while True:  
 print("\nOptions:\n1. Add Student\n2. Update Grade\n3. Show All\n4. Exit")  
 choice = input("Enter choice: ")  
  
 if choice == "1":  
 name = input("Enter student name: ")  
 grade = input("Enter grade: ")  
 grades[name] = grade  
 print(f"Added {name} with grade {grade}")  
  
 elif choice == "2":  
 name = input("Enter student name to update: ")  
 if name in grades:  
 grade = input("Enter new grade: ")  
 grades[name] = grade  
 print(f"Updated {name} to grade {grade}")  
 else:  
 print("Student not found.")  
  
 elif choice == "3":  
 for student, grade in grades.items():  
 print(f"{student}: {grade}")  
  
 elif choice == "4":  
 break  
 else:  
 print("Invalid choice.")

Sample Output:  
Options:  
1. Add Student  
2. Update Grade  
3. Show All  
4. Exit  
Enter choice: 1  
Enter student name: Alice  
Enter grade: A  
Added Alice with grade A  






## 3. Write to a File

This program creates and writes content to a text file using Python file handling functions.  
It uses open() in write mode and the write() function to add text to the file.

Code:

with open("output.txt", "w") as file:  
 file.write("Hello, this is some sample content.\nWelcome to file handling in Python!")  
print("Content written to output.txt")

Sample Output:  
Content written to output.txt

## 4. Read from a File

This program reads and displays content from a text file.  
It uses open() in read mode and the read() function to print the content.

Code:

with open("output.txt", "r") as file:  
 content = file.read()  
 print("File Content:\n", content)

Sample Output:  
File Content:  
Hello, this is some sample content.  
Welcome to file handling in Python!

