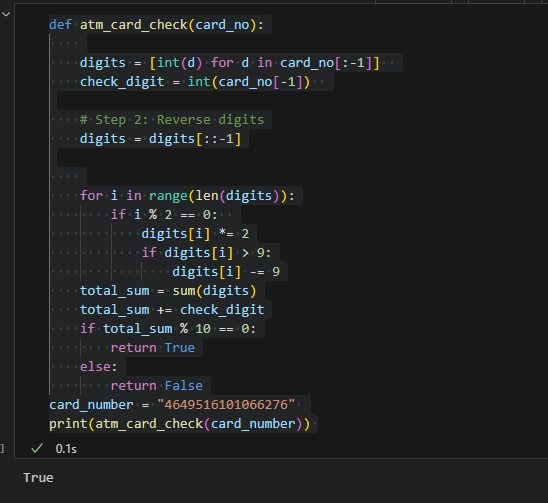
# Q1: ATM Card Check

This function is designed to check whether an ATM card number is valid or not using the Luhn algorithm.  
First, the function receives the card number as input. It separates all digits except the last one into a list of integers. The last digit is stored separately as the check digit.  
Then, the digits list is reversed to follow the Luhn algorithm procedure.  
Next, for each digit in the list, every alternate digit (at even index positions) is doubled. If doubling makes the digit greater than 9, then 9 is subtracted from it to bring it back to a single digit.  
After this transformation, the sum of all digits is calculated. Then, the check digit (the last digit of the card number) is added to the total sum.  
Finally, if the total sum is divisible by 10, the card number is considered valid and the function returns True. Otherwise, it returns False.  
An example card number is tested and the result (True/False) is printed.

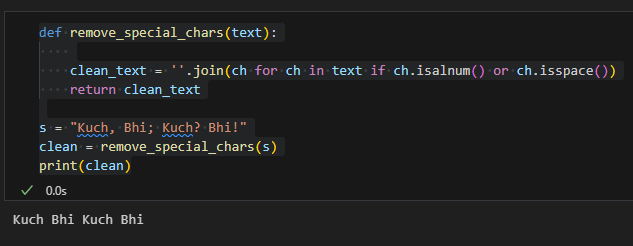
**Output:**

****

# Q2: Remove Special Characters

This function is created to clean a string by removing special characters while keeping only alphabets, numbers, and spaces.  
The function goes through each character in the input string one by one. If the character is alphanumeric (a letter or a digit) or a space, it is kept. Otherwise, it is ignored.  
Finally, all the kept characters are joined back together into a clean string without any punctuation marks or special symbols.  
For example, when given the string 'Kuch, Bhi; Kuch? Bhi!', the output will be 'Kuch Bhi Kuch Bhi'.

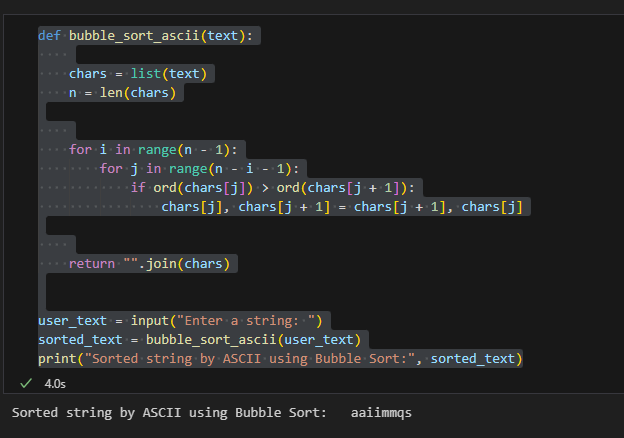
**Output:**

****

# Q3: Bubble Sort ASCII

This function is used to sort the characters of a string based on their ASCII values using the Bubble Sort algorithm.  
The input text is first converted into a list of individual characters. Then, the length of this list is calculated.  
The Bubble Sort algorithm is applied: for each character, it is compared with the next one. If the ASCII value of the current character is greater than the next one, they are swapped. This process continues until the entire list is sorted.  
Finally, the sorted list of characters is joined back into a string and returned.  
For example, if the user inputs a string, the program will sort all its characters in increasing order of their ASCII values.

**Output:**

****