**MACHINE LEARNING –ICP#2**

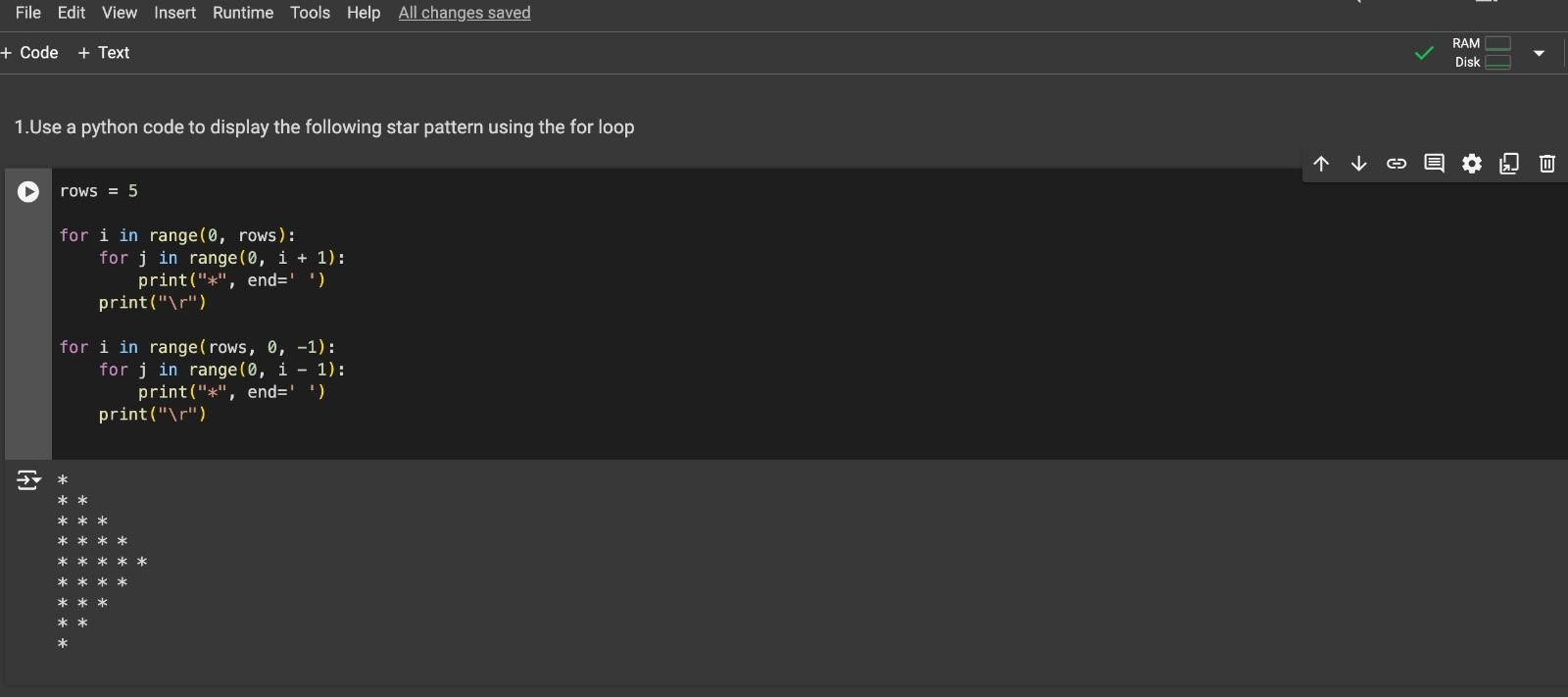
**Student Name** : Manjunath Reddy Velagacherla

**Student Id**: 700759509

**Github Link**: https://github.com/Mxv95090/ICP-2

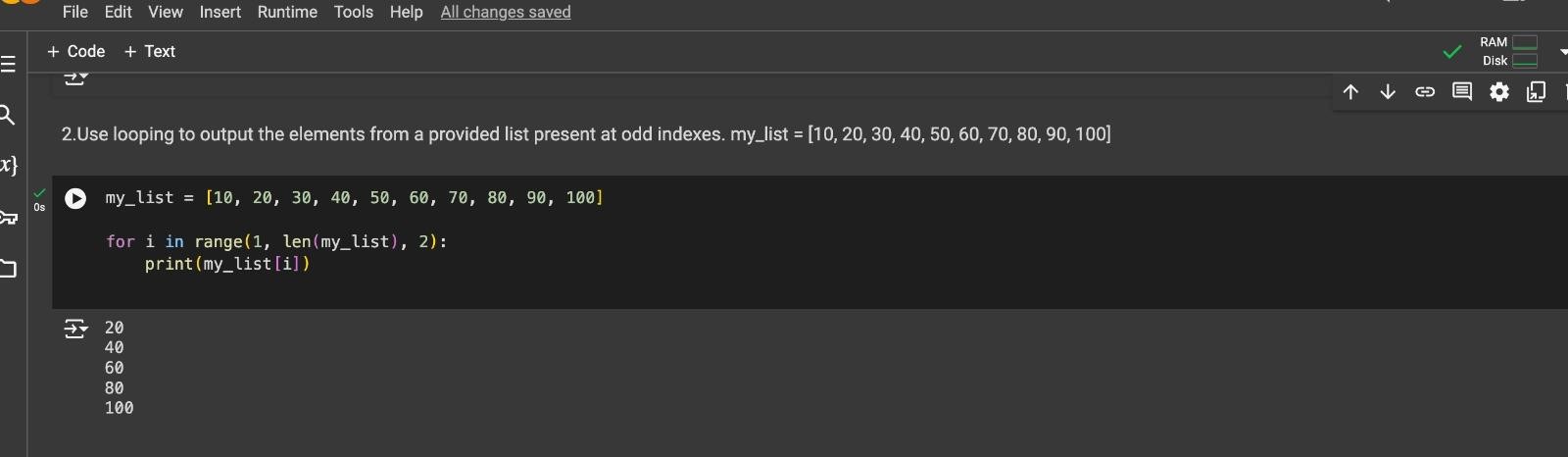
**Question 1:**

The number of rows in the pattern is represented by the variable rows, which is first initialized to 5. After that, the star pattern is printed using two nested for loops. The inner loop prints i+1 stars in each row while the outside loop iterates from 0 to rows-1 in the first half of the pattern. The inner loop prints i-1 stars in each row while the outer loop iterates from rows down to 1 in steps of -1 in the second part of the pattern.



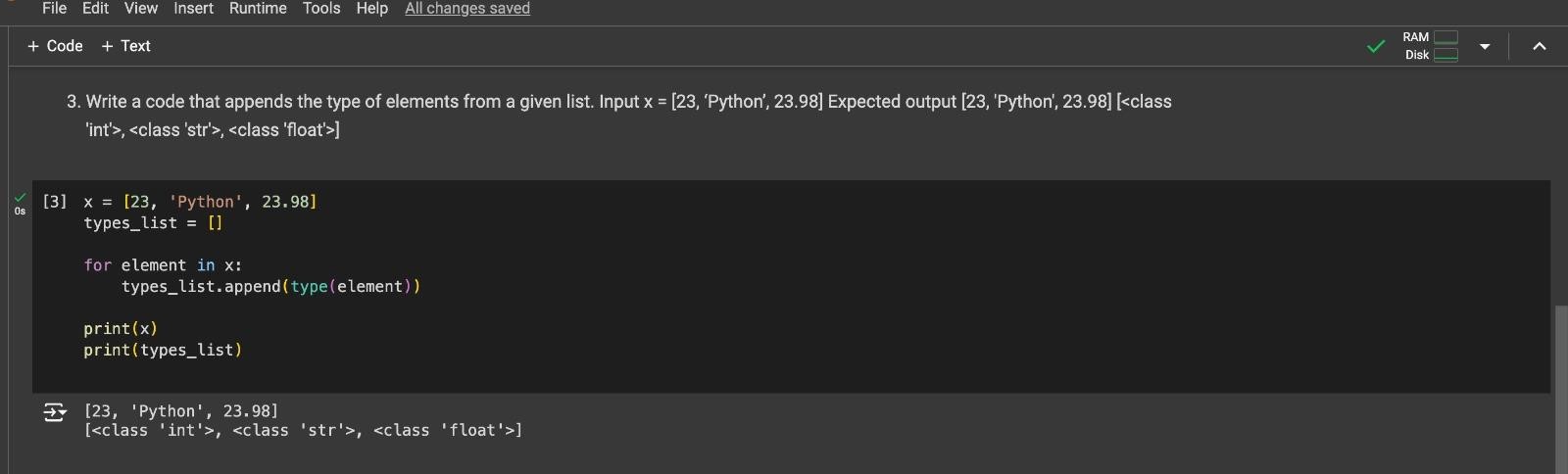
**Question 2:**

The code initializes a list named my\_list containing integers. It uses a for loop with the range() function to iterate over the indices of my\_list. The loop starts from index 1 because it wants to print elements at odd indices. The loop goes up to len(my\_list) (the length of the list) with a step of 2 ensuring it only visits odd indices. Inside the loop, it prints the element at the current odd index using my\_list[i].



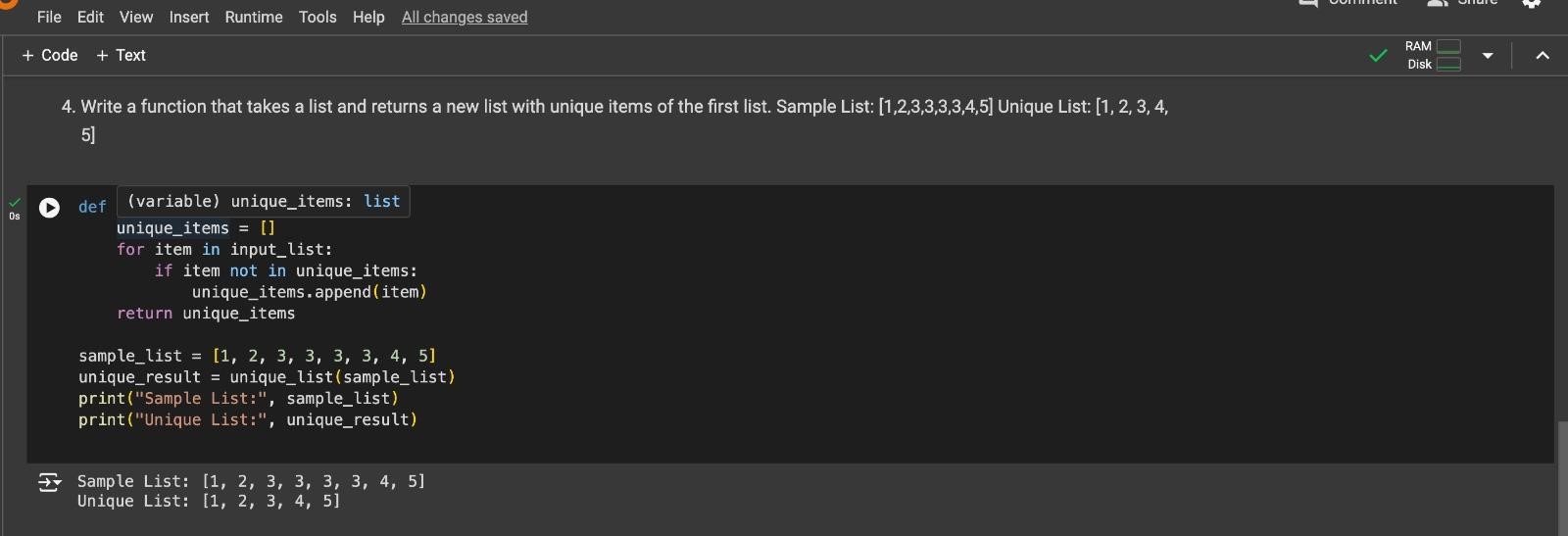
**Question 3:**

We iterate over each element in the list x. For each element, we use the type() function to determine its type, and then append that type to the types\_list. Finally, we print both the original list x and the list containing types types\_list.



**Question 4:**

The function unique\_list takes a list input\_list as input. It initializes an empty list unique\_items to store unique elements. It iterates through each element in the input list. For each element, it checks if it's already in the unique\_items list. If not, it appends it. Finally, it returns the list of unique items.



**Question 5:**

The function count\_case\_characters accepts a string input\_string as input. It initializes variables upper\_count and lower\_count to store the counts of uppercase and lowercase letters, respectively.It iterates through each character in the input string.For each character, it checks if it's uppercase using the isupper() method and increments the upper\_count if it's true. Similarly, it checks if the character is lowercase using the islower() method and increments the lower\_count if it's true. Finally, it returns the counts of uppercase and lowercase characters

