Name: Akash Tamidalapati

Subject: Machine Learning

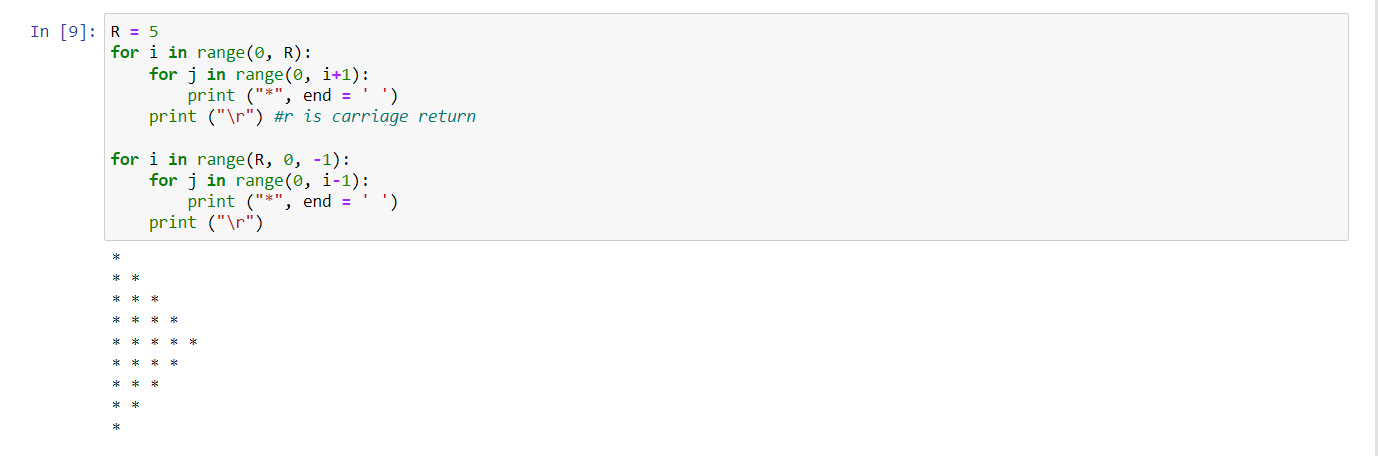
Course ID: CS-5710

Email: axt51210@ucmo.edu

CRN: 31219

Github Link:

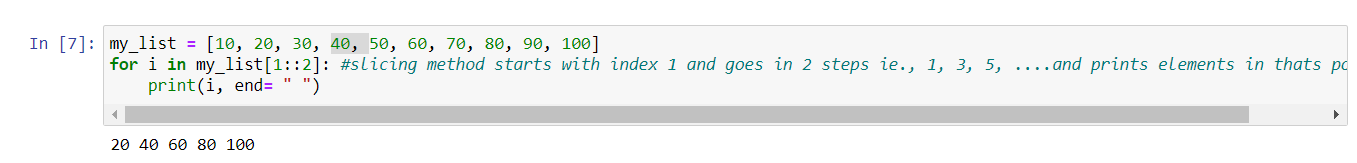
1. Use a python code to display the following star pattern using the for loop



This code generates two nested loops in python that print out a pattern of “\*”. The first loop prints out ascending asterisks and the second loop prints out descending asterisks. The "\r" character is used to indicate a carriage return and start a new line.

1. Use looping to output the elements from a provided list present at odd indexes.

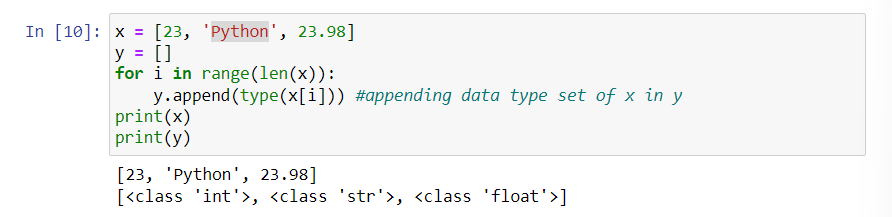
my\_list = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]



This code creates a list of numbers my\_list from 10 to 100, and then uses slicing to print every other number starting from the second element (index 1) in the list. The "end" parameter in the print statement is set to a space to separate the printed numbers with a space instead of a new line.

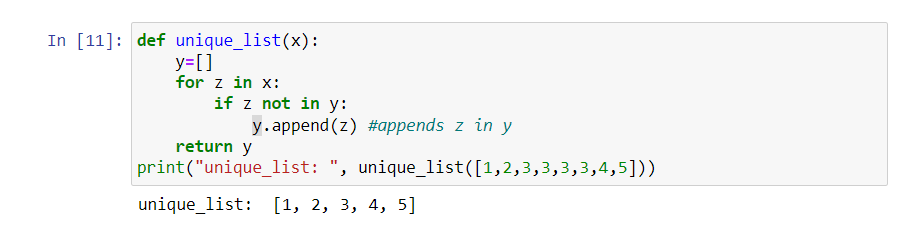
1. Write a code that appends the type of elements from a given list.

Input x = [23, ‘Python’, 23.98] Expected output [23, 'Python', 23.98]



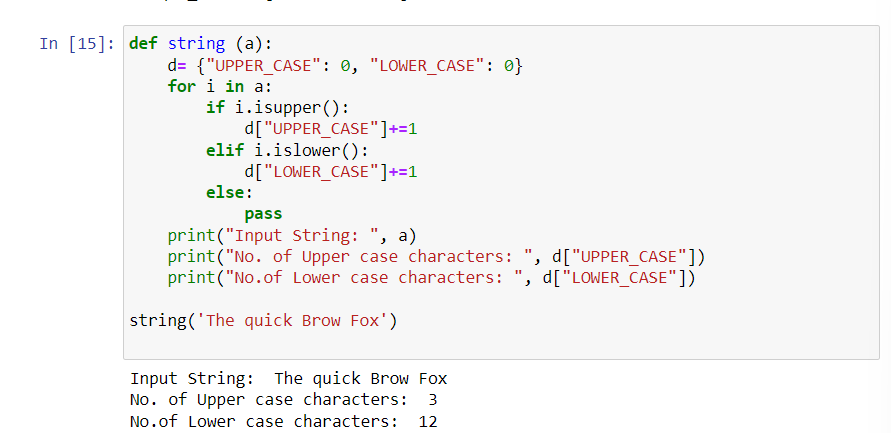
This code creates a list "x" containing elements of different data types (23, “Python”, and 23.98), and then creates an empty list "y". The for loop iterates through each element in "x" and appends its data type to "y". Finally, the code prints both the original list "x" and the list of data types "y".

1. Write a function that takes a list and returns a new list with unique items of the first list. Sample List: [1,2,3,3,3,3,4,5] Unique List: [1, 2, 3, 4, 5]



This code defines a function "unique\_list" that takes a list as an argument and returns a new list that contains only the unique elements from the original list. The function uses a for loop to iterate over the elements in the input list and checks if each element is already in the new list "y". If not, the element is appended to "y". The code then calls the function and passes a list of numbers as an argument, and prints the resulting unique list.

1. Write a function that accepts a string and calculate the number of upper-case letters and lower-case letters. Input String: 'The quick Brow Fox' Expected Output: No. of Upper-case characters: 3 No. of Lower-case Characters: 12



This code defines a function "string" that takes a string as an argument and calculates the number of upper-case and lower-case letters in the input string. The function initializes a dictionary "d" with keys "UPPER\_CASE" and "LOWER\_CASE" and values 0. The for loop iterates over each character in the input string and increments the count of upper-case or lower-case letters in "d" depending on the case of the character. The code then prints the input string and the number of upper-case and lower-case letters in the string. The function is then called with "The quick Brow Fox".