# **Assignments: For Loops Learning Path**

## **1. Counting in a Loop**

**Objective:** Understand the basic structure of a for-loop.  
**Task:**  
Write a program that prints the numbers 1 through 10 on separate lines.

**Guidance for students:**

* Think about what three components go inside a for-loop: *initialization*, *condition*, and *increment*.
* Ask yourself: What’s the starting number? When should it stop? By how much should it step?

## **2. Summing Numbers**

**Objective:** Explore how for-loops can calculate and store values.  
**Task:**  
Write a program that finds the sum of numbers from 1 to 100.

**Guidance:**

* Instead of printing, store results in a variable (e.g., total).
* Add the loop variable to this running total each iteration.
* At the end, print the total.
* Question to consider: How does your result connect to the arithmetic series formula?

## **3. Patterns with Stars**

**Objective:** Practice nested for-loops.  
**Task:**  
Write a program that prints a right-angled triangle made of \*, like this (5 rows):

\*  
\*\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*

**Guidance:**

* A loop helps print each row. Another loop controls how many stars in the row.
* Think: The row number tells you how many times to print \*.
* Challenge extension: Invert the triangle.

## **4. Multiplication Table**

**Objective:** Apply loops to generate structured output.  
**Task:**  
Write a program that prints the multiplication table (1 through 10).

**Guidance:**

* Use **nested loops**: one for the rows, one for the columns.
* Decide: Should you print results aligned in a grid?
* Extension question: How can you format the table neatly using \t (tab) or string formatting?

## **5. Word Analyzer**

**Objective:** Apply loops to process text (strings).  
**Task:**  
Write a program that takes a word or sentence as input, then:

1. Counts the number of vowels in it.
2. Reverses the word/sentence using a loop.
3. Prints the result in both normal and reversed forms, along with the vowel count.

**Guidance:**

* Remember: a string can be *treated like a sequence*, so you can loop over its characters.
* Ask: How do you check if a character is a vowel? (Hint: membership test "a" in "aeiou").
* To reverse, loop backwards from the last character to the first.
* Bonus thinking: How would it change if we also counted words, not just letters?