**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Partner: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Python Activity 9: Looping Structures: FOR Loops**

|  |
| --- |
| **Learning Objectives**  Students will be able to:  *Content:*   * Explain the difference between **while loop(conditioned controlled)**  and a **FOR loop(count controlled\*\*HARDER\_IN\_PYTHON)** * Explain the syntax of a **FOR loop** * Explain how to use the **range()** function in a **FOR loop** * Explain an **accumulator**  in a **FOR loop**   *Process:*   * Write code that includes **FOR loop** * Write code that uses use **FOR loops** within functions   **Prior Knowledge**   * Python concepts from Activities 1-8 |

**Critical Thinking Questions:**

1. Enter and execute the following two Python programs.

|  |
| --- |
| **WHILE LOOP -- Python Program** |
| **FOR LOOP – Python Program** |

a. What is the output for each program?

\_\_\_\_\_\_\_\_\_\_\_\_\_Both print the input for their conditioned output (this case it’s 20) \_\_\_\_\_

b. Both programs produce the same output. Which code fragment is more concise?

\_\_\_\_\_\_\_\_\_\_\_\_\_The “for” loop; as you can specify the range/amount of times it loops.

|  |
| --- |
| **FYI:** The Python predefined function - **range()** - is used to define a series of numbers and can be used in a FOR loop to determine the number of times the loop is executed.. |

2. Enter and execute the following code fragments and state the output:

a. for x in range(5):

print(x, end=" ") \_\_\_\_\_\_\_0 1 2 3 4 \_\_\_\_\_

b. for x in range(1,5):

print(x, end=" ") \_\_\_\_\_\_\_1 2 3 4

c. for x in range(3,20,2):

print(x, end=" ") \_\_\_\_\_\_3 5 7 9 11 13 15 17 19\_\_\_\_

d. numIterations = 6

for x in range(numIterations):

print(x, end=" ") \_\_\_0 1 2 3 4 5 \_\_\_\_

e. numIterations = 6

for x in range(1, numIterations+1):

print(x, end=" ") \_\_\_\_1 2 3 4 5\_\_\_\_\_\_\_

3. After examining the five code fragments in #2, explain how the **range()** function works. Include an explanation of the arguments.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_For a range, when you are only given one variable inside the “()” it will include every number BEFORE said variable. You can even pick where it starts by adding the first integer followed by a comma and the end integer: EX:((1,21)= 1 2 3 4 5 6….). For the last variable, you can change what increments it outputs each integer leading UP to the limiter. EX:((1, 21, 3) = 1 4 7…)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| **FYI:** In a FOR loop you can include a list of values in place of the **range()** function. |

4. Enter and execute the following code.

for x in [3,6,9,12,15,18]:

print(x, end=” “)

a. Rewrite this code using the **range()** function.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_range( 3, 18 ,3)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b. Why would you use the **range()** function when you could just list the numbers?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_It is more efficient\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| **FYI:** The **str()** function converts what is the parentheses ( ) to a String. |

5. Read through the code and determine what it does.

**favorite = input("Enter your favorite ice cream flavor: ")**

**for x in range(1,5):**

**print(str(x) + “.”, favorite, end="\t")**

a. What do you think the program does? \_\_\_\_\_\_\_It lists out your favorite icecream that you just input \_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b. Enter and execute the code to determine if you were correct. What does the program actually do? Provide a detailed explanation.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_It lists your input from range 1, to 4, as it ends in five, and only uses your input as it doesn’t ask you how many flavors you would list.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

c. Why is the **str()** function needed in the print statement?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_As it will give a syntax error if you do not specify the “X” in favorite with anything. \_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6. Complete the arguments in the following range function so that the code prints the even numbers between 100 and 200 inclusive.

**for x in range(100, 200, 2):**

**print(x)**

7. Complete the arguments in the following range function so that the code prints: 5 4 3 2 1 0.

**for x in range(5, -1, -1):**

**print(x, end=” “)**

|  |
| --- |
| **FYI:** An **accumulator** is a variable that stores the sum of a group of values. |

8. Examine the following code segment.

total = 0

for x in range(5):

number = int(input("Enter a number: "))

total += number

print("The total is:",total)

a. Why is the variable **total** initialized to 0 in the first line of code?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_So that it adds your input to 0\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b. Explain what the following code does:

**number = int(input("Enter a number: "))**

\_\_\_\_\_\_\_\_\_This asks for an input from the user; that it gives an integer format. \_\_\_

c. Explain what the following code does: **total += number**

\_\_\_\_\_\_\_\_\_\_\_\_\_It is adding “number” to the output continuously until the statement is no longer true \_

d. How many numbers does the program prompt for? \_\_\_\_\_One\_\_\_\_\_

e. What is the **accumulator** in the code segment? \_\_\_\_\_ total = 0\_\_\_\_\_\_\_\_

9. Is it better to use a **FOR loop** when you know the number of times the loop should be executed or when you do not know? \_\_\_\_\_\_\_\_\_\_\_\_When you know\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Application Questions: Use the Python Interpreter to check your work**

1. Write a code segment using a FOR loop that prints multiples of 5 from 5 to 500, one on a line.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_