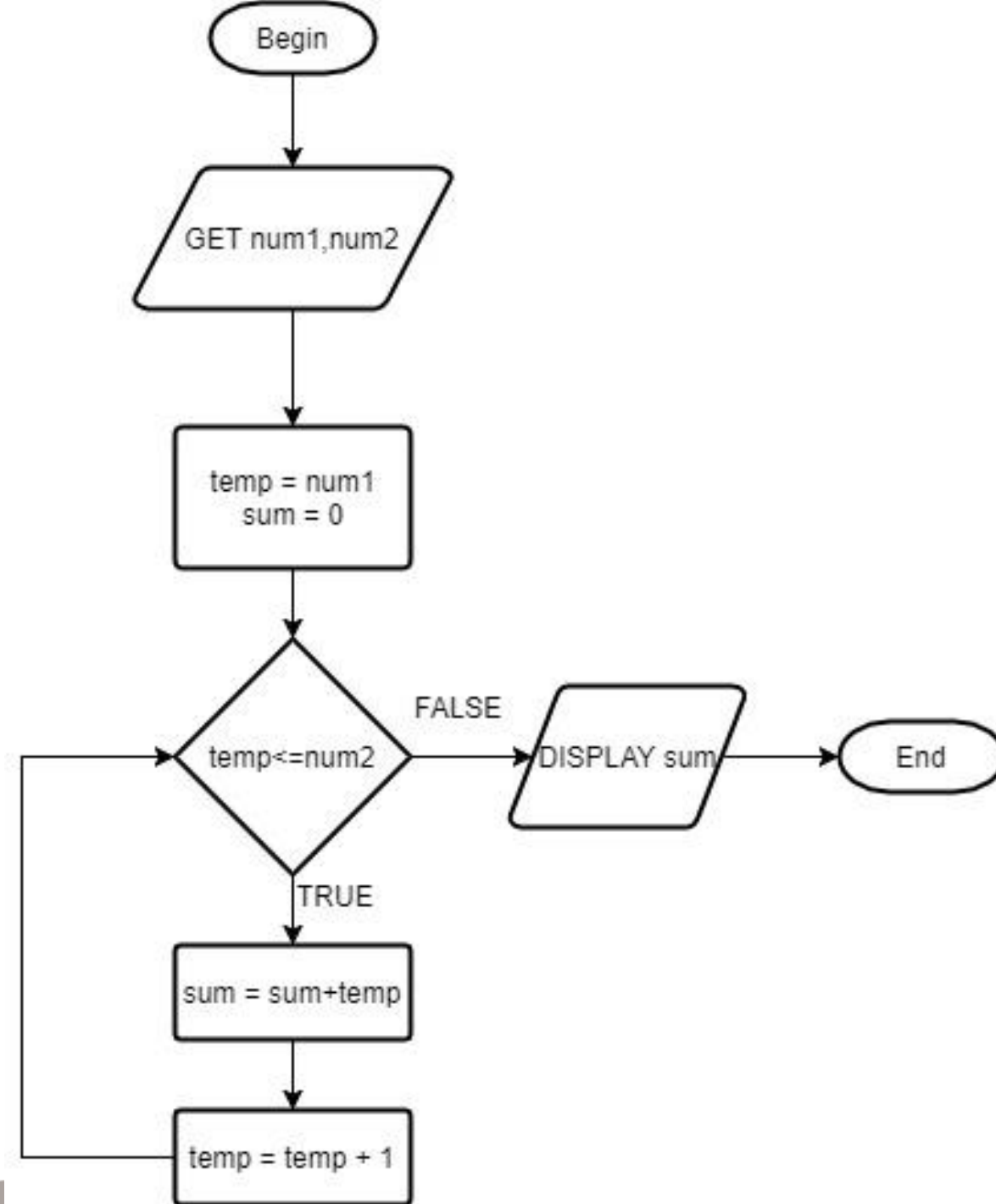


# 953103 Programming Logic Thinking

For loop

# Problem Set

*Write a program to receive 2 numbers from users, calculate the summation of the numbers in the range and display summation.*



Screen1

Number 1

Number 2

Result

initialize global num1 to 0

initialize global num2 to 0



initialize global temp to 0

initialize global sum to 0

when Button2 .Click

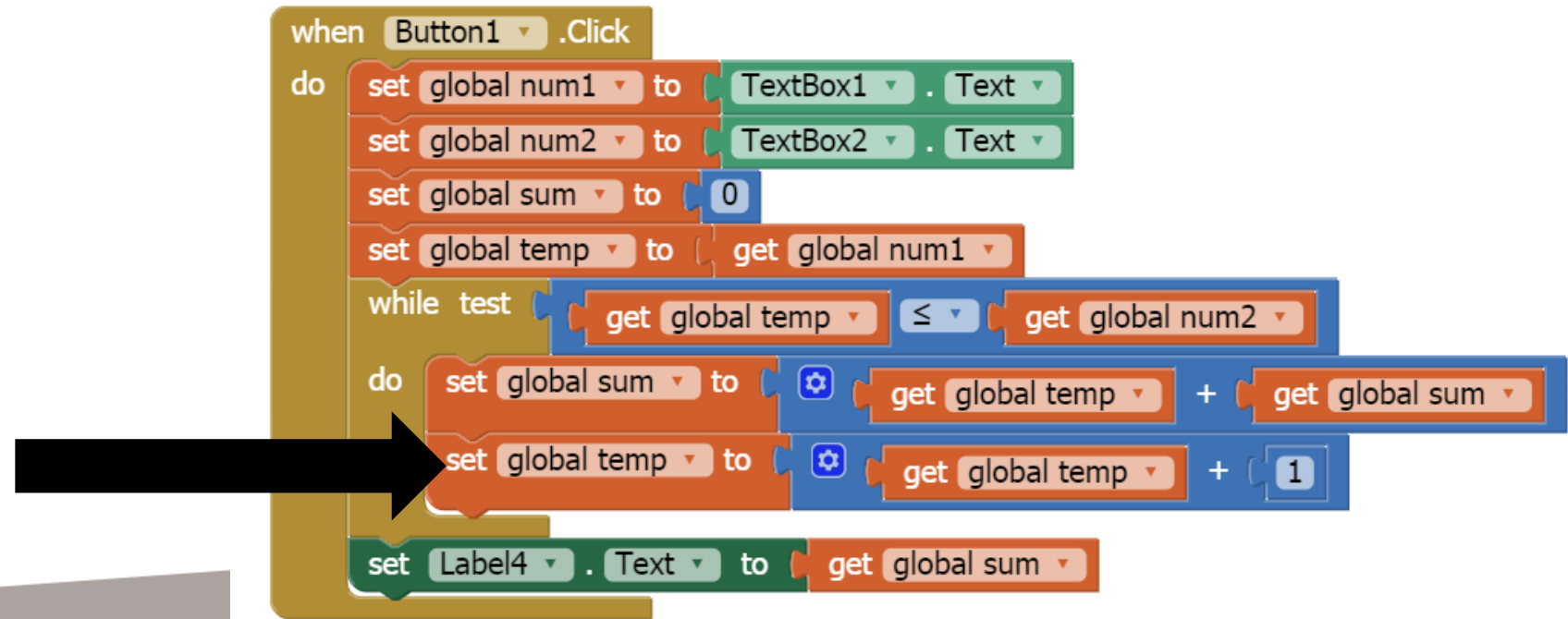
do  
  set TextBox1 . Text to 0  
  set TextBox2 . Text to 0

when Button1 .Click

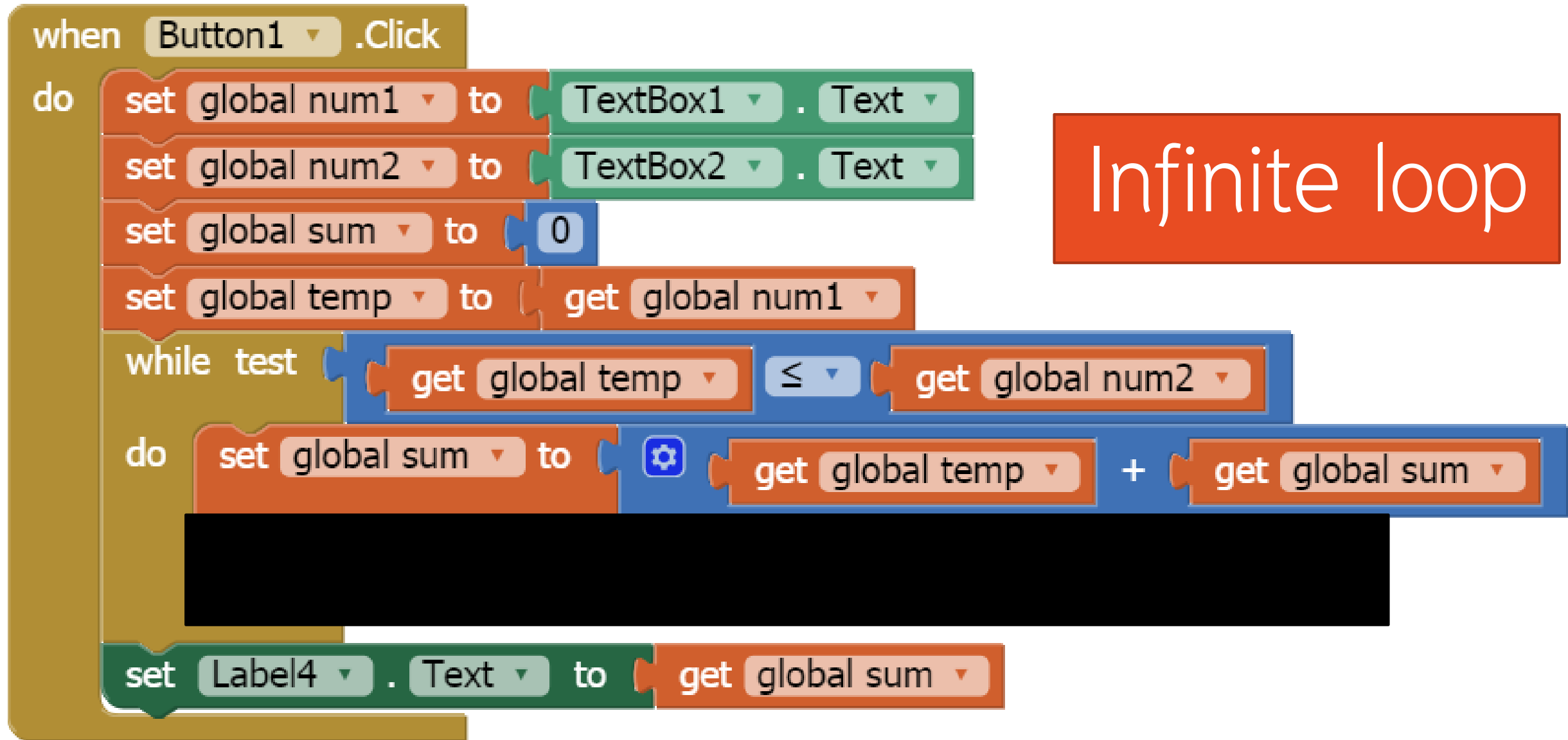
do  
  set global num1 to TextBox1 . Text  
  set global num2 to TextBox2 . Text  
  set global sum to 0  
  set global temp to get global num1  
  while test  
    get global temp ≤ get global num2  
  do  
    set global sum to  get global temp + get global sum  
    set global temp to  get global temp + 1  
  set Label4 . Text to get global sum

# Counter variable

- The counter variable
  - The variable is used to count the integrations.
  - The value is increased by each iteration.
- The counter variable is often located at the end of the body.
- The incremental process is called *update statement*.



What would happen if there is  
no update statement?



Infinite loop



# Update statement

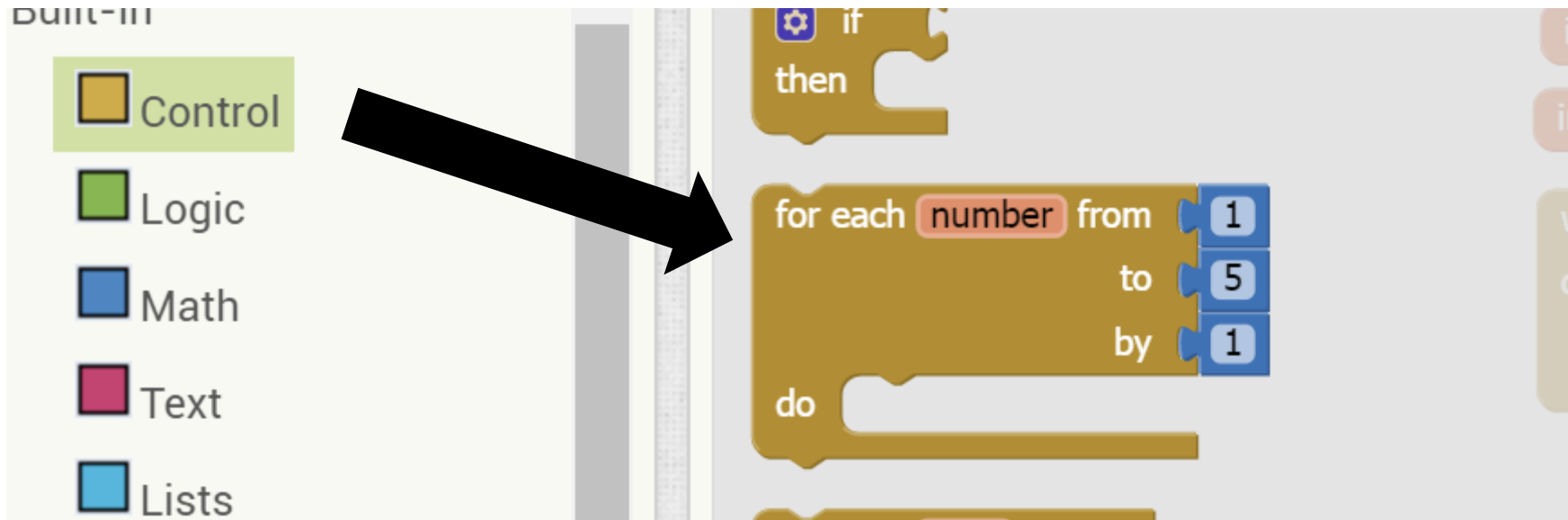
- The purpose of the update statement is to make the iteration to move forward.
  - The progress of the iteration is not limited to 1. It can be 2 or any other number.
  - The progress of the iteration can also be decreased. (count down)

# Fixed Iteration Loop

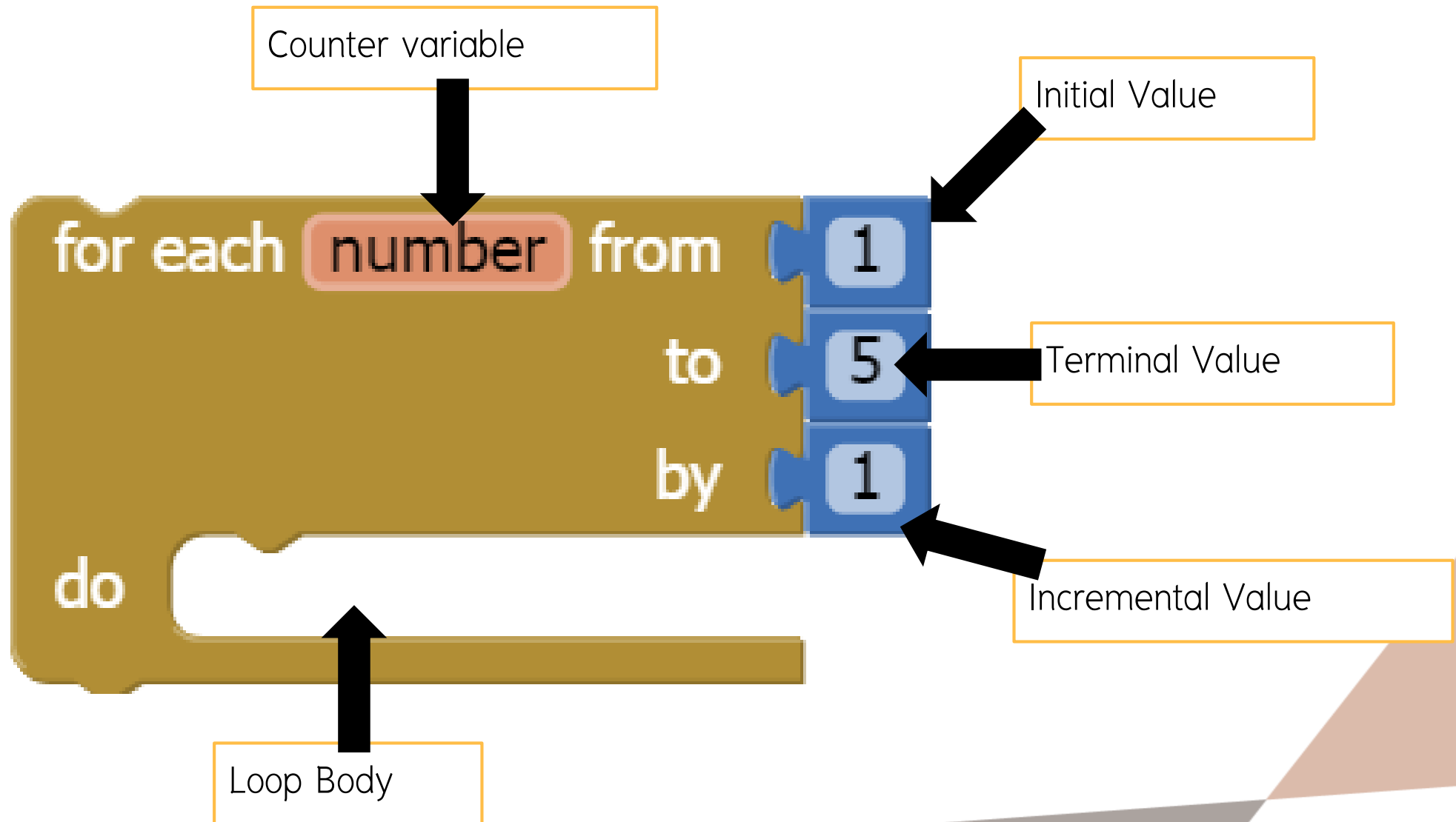
- Fixed Iteration Loop
  - Do something with the specific amount of iteration.
  - The counter variable keeps track of the iteration.
    - When an iteration passes, the counter variable increases.
- In programming language, the fixed iteration loop is called
  - FOR-loop

# The for Statement

- The syntax of the for statement is:



- The braces may be omitted if the statement block consists of a *single* statement.



# The for Statement

For each time, FOR statement executes:

1. *Initialization process*. The counter variable is generated and is set to the initial value.
2. *Repetition process*. The condition is evaluated.

Counter variable reaches Terminal number

2.1 If the loop condition is *true*, then:

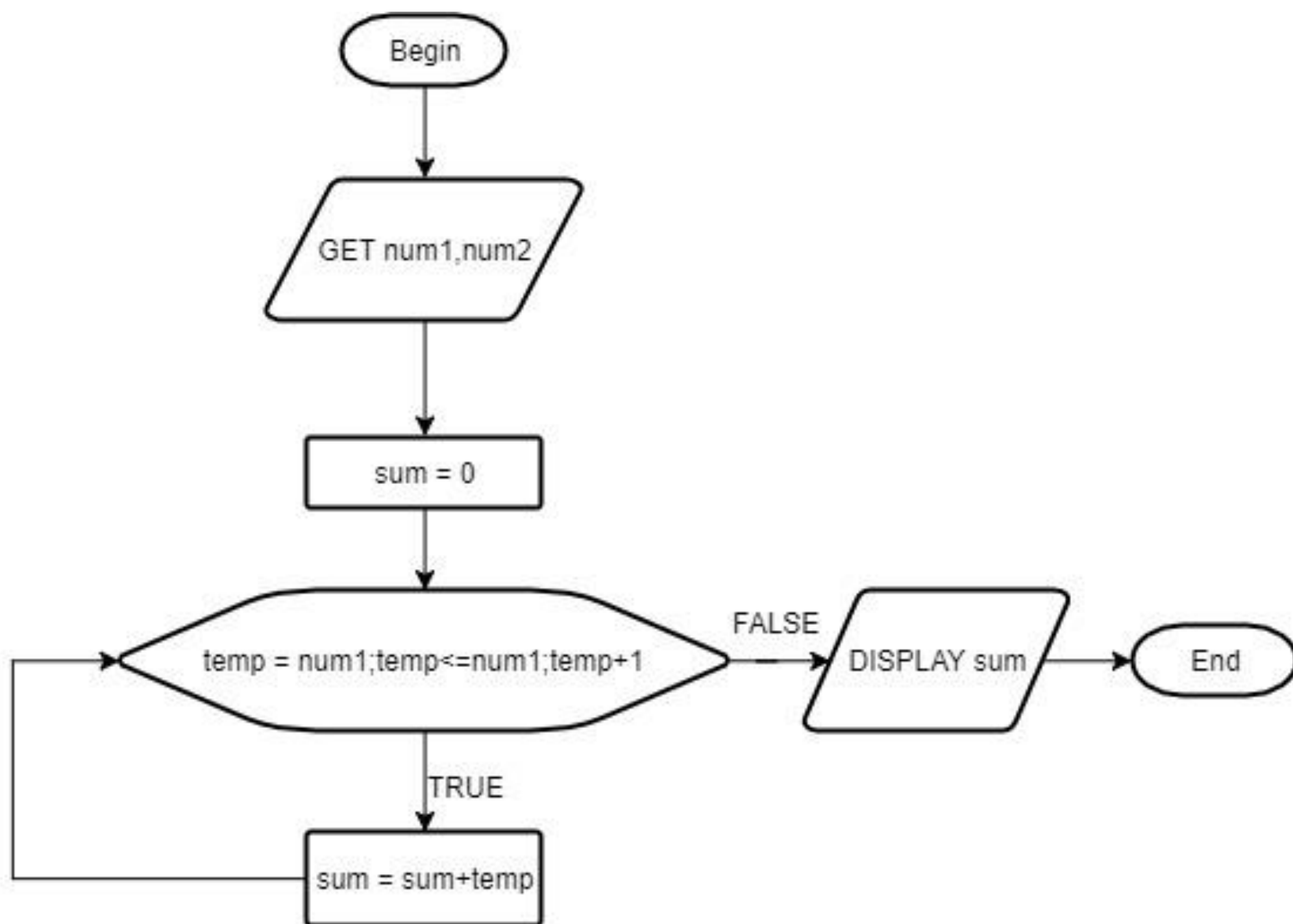
2.1.1 statement-1, statement-2,..., statement-n execute,

2.1.2 Update statement : control variable is increased by incremental value.

2.1.3 Go to step 2.

2.2 If the loop condition is *false*

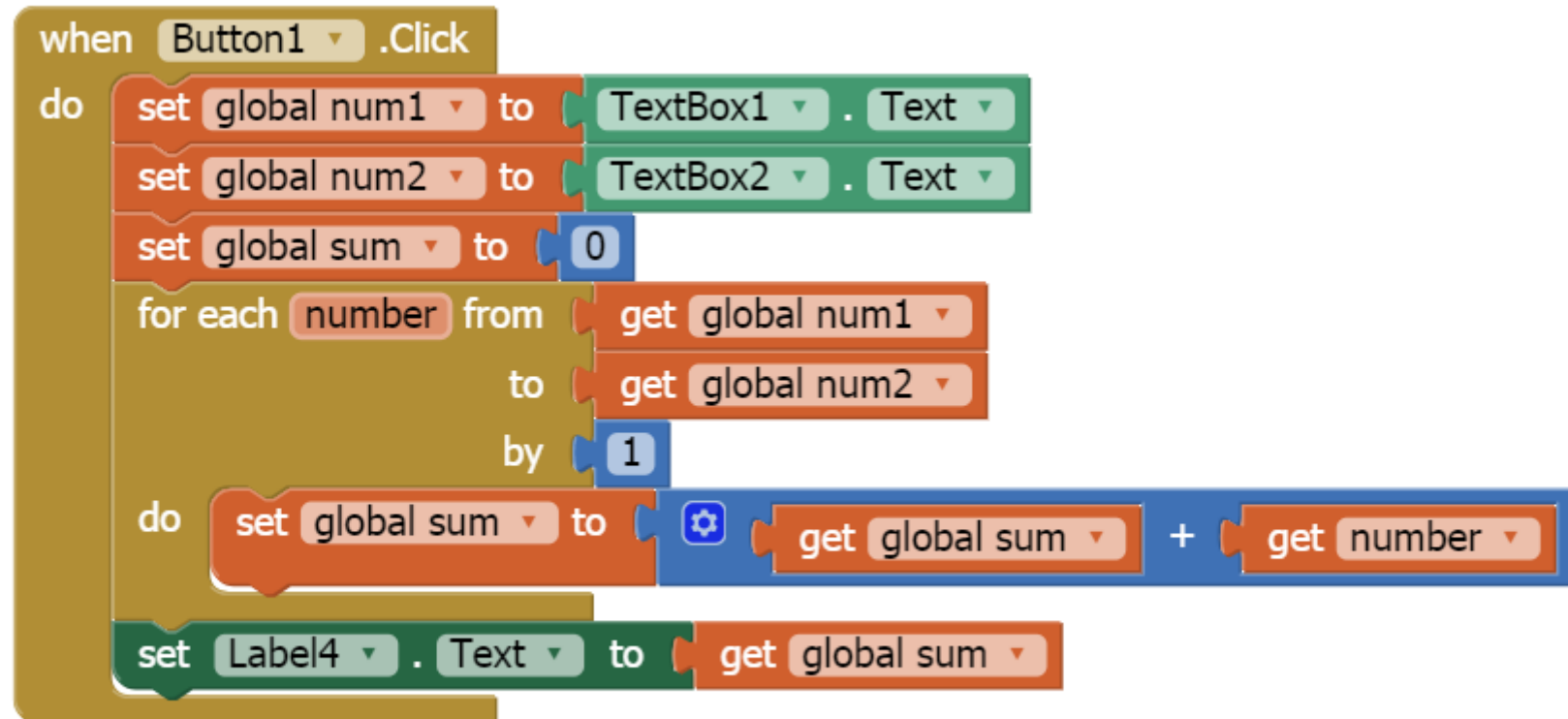
2.2.1 The loop is terminated.



initialize global temp to 0

```
when Button1 .Click
do
  set global num1 to TextBox1 . Text
  set global num2 to TextBox2 . Text
  set global sum to 0
  set global temp to get global num1
  while test
    get global temp ≤ get global num2
  do
    set global sum to get global temp + get global sum
    set global temp to get global temp + 1
  set Label4 . Text to get global sum
```

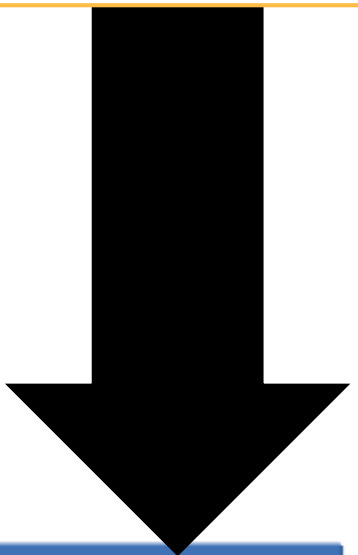
```
for each number from 1
to 5
by 1
do
```





```
when Button1 .Click
do
  set global num1 to TextBox1 . Text
  set global num2 to TextBox2 . Text
  set global sum to 0
  set global temp to get global num1
  while test
    get global temp ≤ get global num2
  do
    set global sum to get global temp + get global sum
    set global temp to get global temp + 1
  set Label4 . Text to get global sum
```

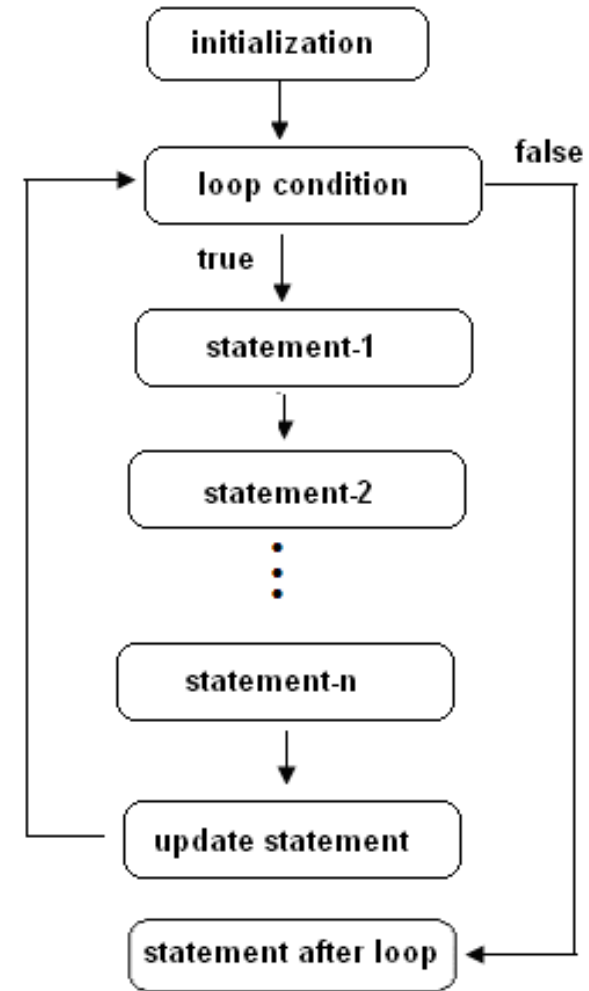
The counter variable generated by the FOR loop.



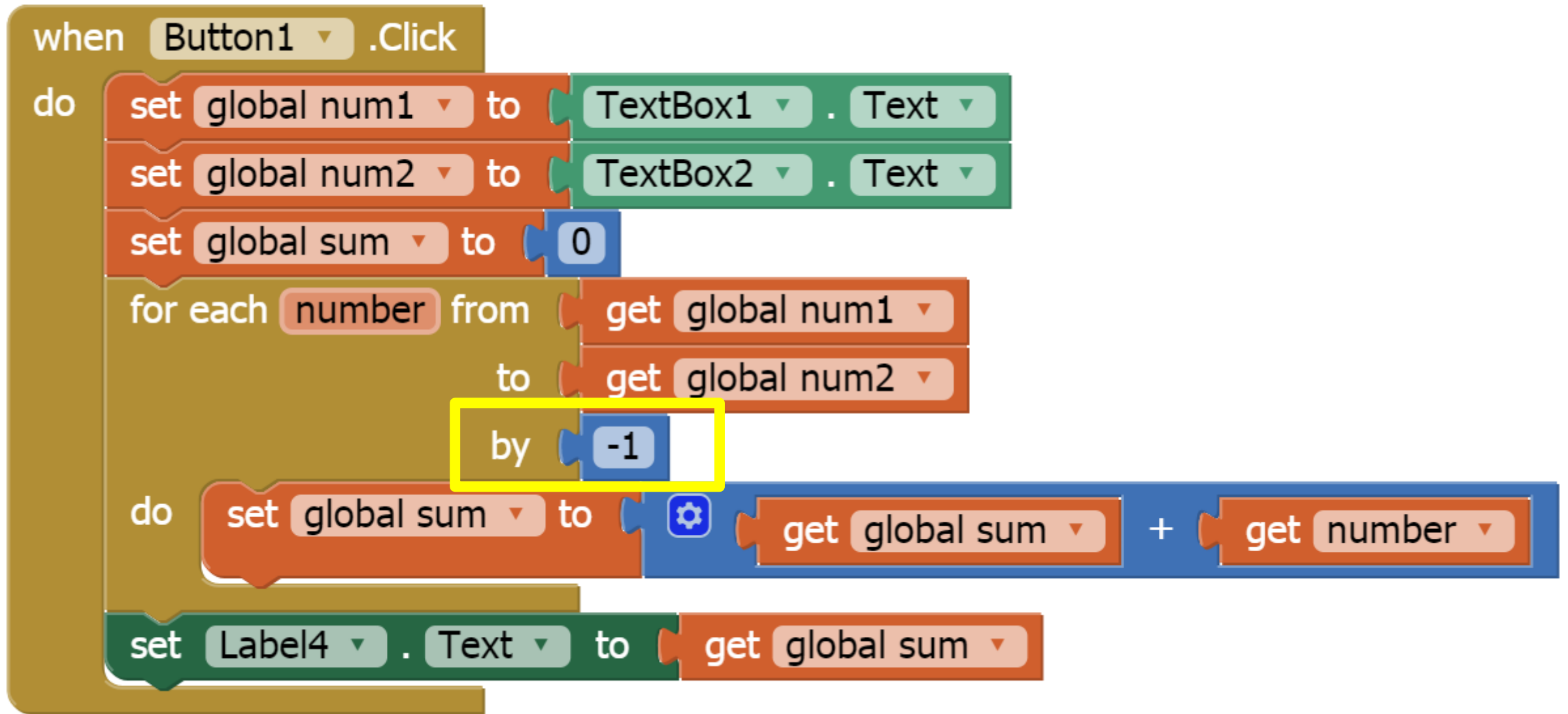
```
when Button1 .Click
do
  set global num1 to TextBox1 . Text
  set global num2 to TextBox2 . Text
  set global sum to 0
  for each number from
    get global num1
    to get global num2
    by 1
  do
    set global sum to get global sum + get number
  set Label4 . Text to get global sum
```

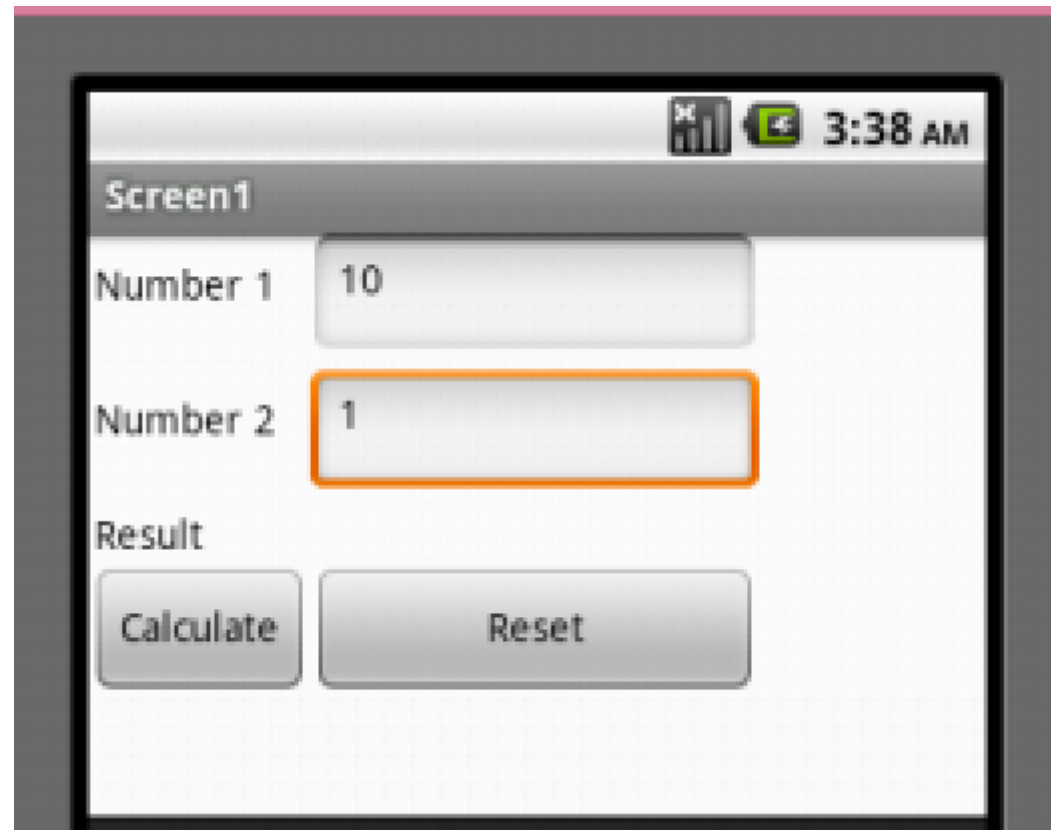
# The for Statement

- The initialization is performed *exactly once*.
- The loop condition is *always* tested before the statement block *executes*.
- The update statement always executes *after* the actions of the statement block.
- The declared, initialized variables *disappear* after the for loop completes execution.



# Case Study





What is the result?

Screen1

Number 1 10

Number 2 1

Result 55

Calculate Reset

# Another Case Study



Screen1

Number 1

Number 2

Result

What is the result?

Screen1

Number 1 1

Number 2 10

Result 0

Calculate Reset