Programming Fundamentals II

Lab 5: Loops

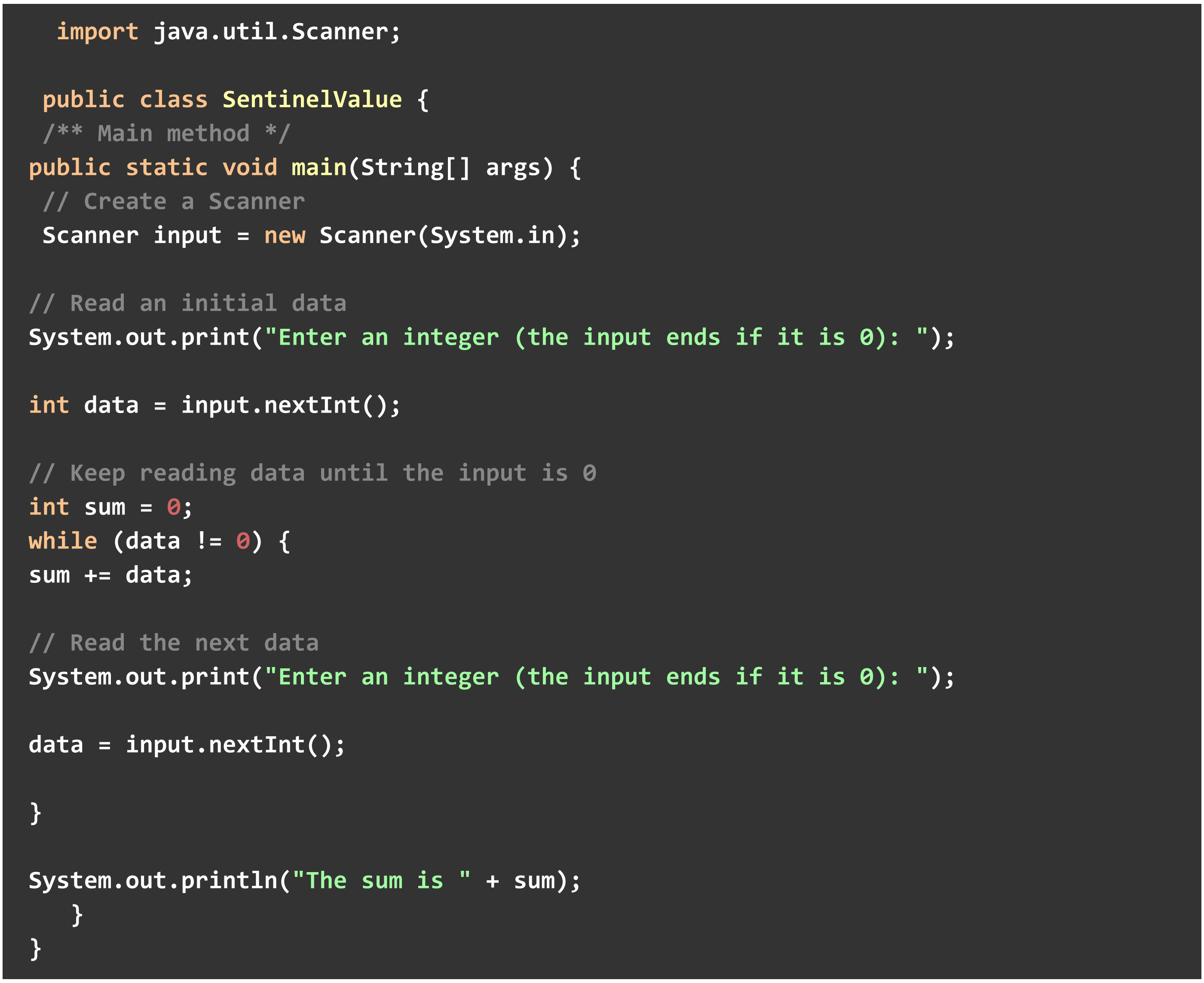
Learning Objectives:

- Write programs that execute statements repeatedly using a **while** loop.
- Develop loops using **do-while** statements.
- Construct loops with **for** statements.
- Apply the loop design strategy to create efficient loops.
- Manage program control using break and continue statements.
- Control loop execution with user confirmation or a sentinel value.
- Implement nested loops in your programs.

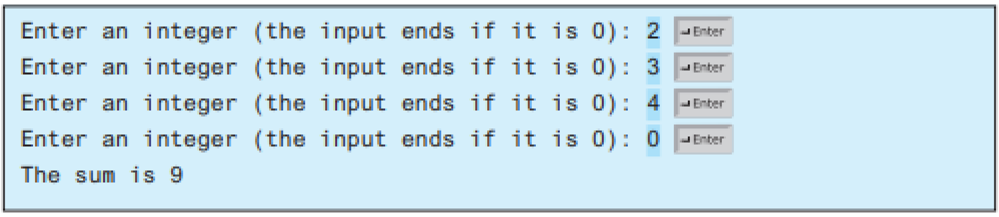
Requisite knowledge:

- Chapter 5 Elementary Programming from “Introduction to Java Programming Brief Version” Reference (See lms.uofk.edu).
- Lecture 5 (See lms.uofk.edu).

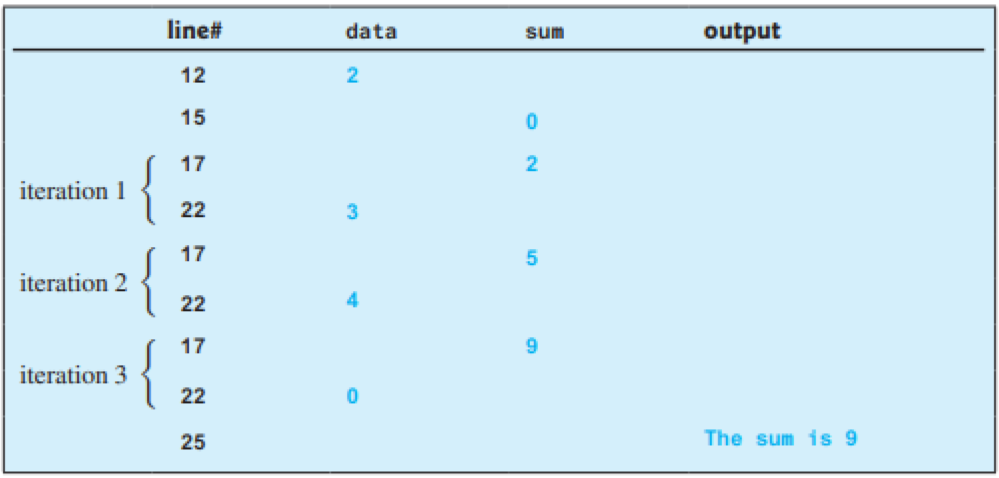
Case Study 1 : SentinelValue



```
  import java.util.Scanner;

 public class SentinelValue {
 /** Main method */
public static void main(String[] args) {
 // Create a Scanner
 Scanner input = new Scanner(System.in);

// Read an initial data
System.out.print("Enter an integer (the input ends if it is 0): ");

int data = input.nextInt();

// Keep reading data until the input is 0
int sum = 0;
while (data != 0) {
sum += data;

// Read the next data
System.out.print("Enter an integer (the input ends if it is 0): ");

data = input.nextInt();

}

System.out.println("The sum is " + sum);
   }
}
```



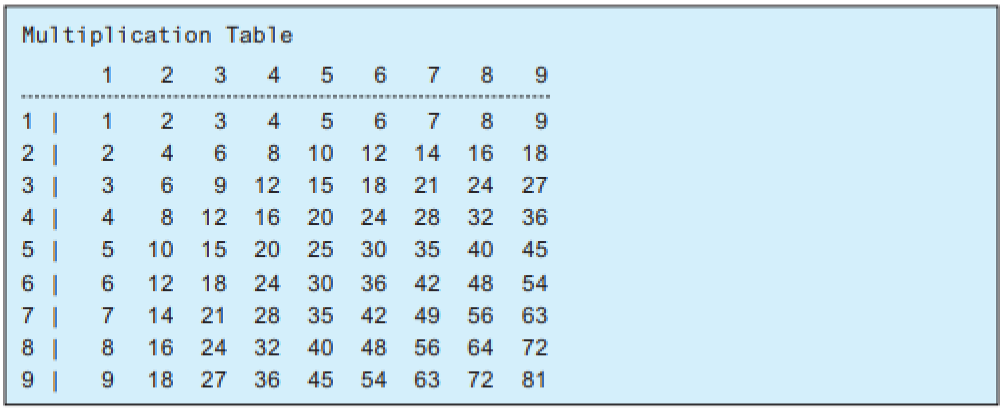
```
Enter an integer (the input ends if it is 0): 2
Enter an integer (the input ends if it is 0): 3
Enter an integer (the input ends if it is 0): 4
Enter an integer (the input ends if it is 0): 0
The sum is 9
```



	line#	data	sum	output
	12	2		
	15		0	
iteration 1	17		2	
	22	3		
iteration 2	17		5	
	22	4		
iteration 3	17		9	
	22	0		
	25			The sum is 9

Case Study 2 : MultiplicationTable

```
public class MultiplicationTable {
 /** Main method */
 public static void main(String[] args) {
 // Display the table heading
System.out.println(" Multiplication Table");

// Display the number title
System.out.print(" ");
 for (int j = 1; j <= 9; j++)
System.out.print(" " + j);11
 System.out.println("\n -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- ---- -- -- --
-- -- -- -- -- -- -- -- --");

// Display table body
for (int i = 1; i <= 9; i++) {
System.out.print(i + " | ");
for (int j = 1; j <= 9; j++) {
// Display the product and align properly
System.out.printf("%4d", i * j);
}
System.out.println();

       }
  }

}
```

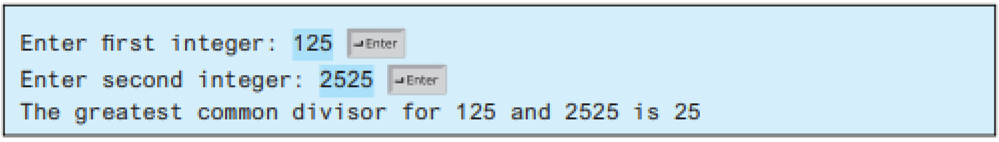


```
Multiplication Table
        1    2    3    4    5    6    7    8    9
-------------------------------------------------
1 |     1    2    3    4    5    6    7    8    9
2 |     2    4    6    8   10   12   14   16   18
3 |     3    6    9   12   15   18   21   24   27
4 |     4    8   12   16   20   24   28   32   36
5 |     5   10   15   20   25   30   35   40   45
6 |     6   12   18   24   30   36   42   48   54
7 |     7   14   21   28   35   42   49   56   63
8 |     8   16   24   32   40   48   56   64   72
9 |     9   18   27   36   45   54   63   72   81
```

Case Study 3 : GreatestCommonDivisor

```
import java.util.Scanner;

 public class GreatestCommonDivisor {
  /** Main method */
  public static void main(String[] args) {
    // Create a Scanner
    Scanner input = new Scanner(System.in);

   // Prompt the user to enter two integers
   System.out.print("Enter first integer: ");
   int n1 = input.nextInt();
   System.out.print("Enter second integer: ");
   int n2 = input.nextInt();

   int gcd = 1; // Initial gcd is 1
   int k = 2; // Possible gcd
   while (k <= n1 && k <= n2) {
     if (n1 % k == 0 && n2 % k == 0)
```

```
           gcd = k; // Update gcd
            k++;
 }
  System.out.println("The greatest common divisor for " + n1 + 24 " and " + n2 + "
is " + gcd);

   }

 }
```

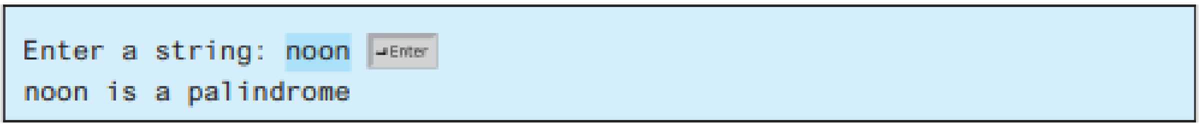


```
Enter first integer: 125
Enter second integer: 2525
The greatest common divisor for 125 and 2525 is 25
```

Case Study 4 : Palindrome

```
import java.util.Scanner;

public class Palindrome {
 /** Main method */
 public static void main(String[] args) {
  // Create a Scanner
  Scanner input = new Scanner(System.in);

  // Prompt the user to enter a string
 System.out.print("Enter a string: ");
 String s = input.nextLine();

 // The index of the first character in the string
 int low = 0;

 // The index of the last character in the string
 int high = s.length() - 1;

 boolean isPalindrome = true;
 while (low < high) {
```

```
if (s.charAt(low) != s.charAt(high)) {
isPalindrome = false;
break;
}

  low++;
  high--;
}

if (isPalindrome)
System.out.println(s + " is a palindrome");
else
System.out.println(s + " is not a palindrome");
  }
}
```



```
Enter a string: noon ↵Enter
noon is a palindrome
```

End Of Lab