COMP 110-001
Loop Statements

Yi Hong

May 21, 2015

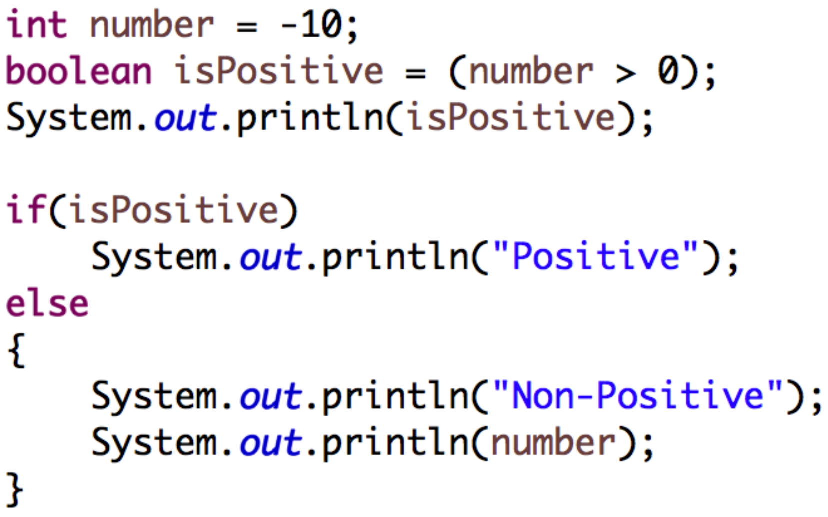
Announcements

- Grades of Lab 0 were posted

Review

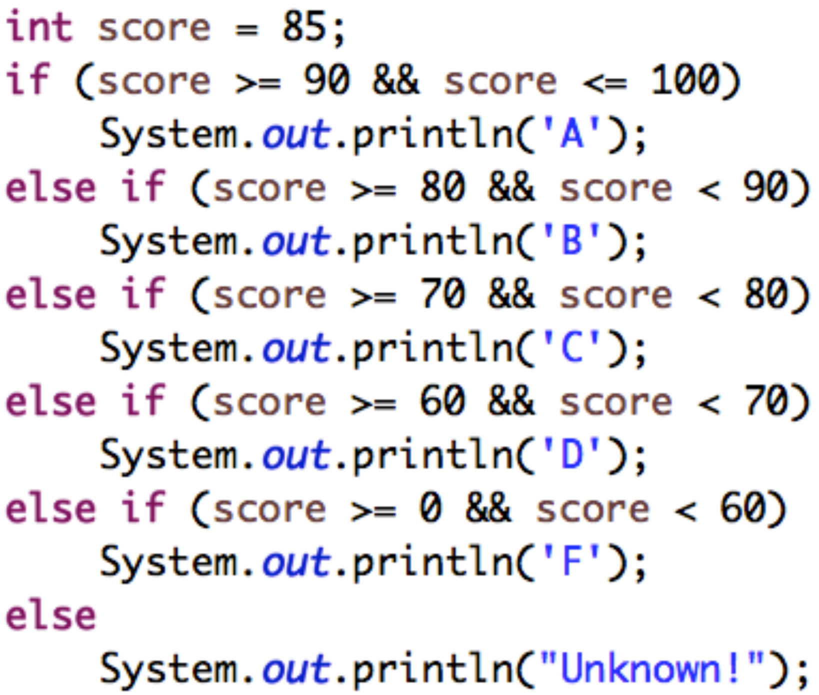
- Q1: What’s the outputs of following statements?

- Q2: Write a program that assigns grade based on an input score
  - A: 90 ~ 100
  - B: 80 ~ 89
  - C: 70 ~ 79
  - D: 60 ~ 69
  - F: 0 ~ 59



```
int number = -10;
boolean isPositive = (number > 0);
System.out.println(isPositive);

if(isPositive)
    System.out.println("Positive");
else
{
    System.out.println("Non-Positive");
    System.out.println(number);
}
```



```
int score = 85;
if (score >= 90 && score <= 100)
    System.out.println('A');
else if (score >= 80 && score < 90)
    System.out.println('B');
else if (score >= 70 && score < 80)
    System.out.println('C');
else if (score >= 60 && score < 70)
    System.out.println('D');
else if (score >= 0 && score < 60)
    System.out.println('F');
else
    System.out.println("Unknown!");
```

If-Else or Switch Statement

- Use a switch statement when you have more than 2 conditions on a single variable
  - Example: Weekdays – if you have a different action to perform for each day of the week, use a switch statement
- Use an if-else for all other scenarios:
  - More than one variable you're testing (multiple conditions)
  - Testing for a range of values
  - Variable is not an int, char, or enum
  - Example: Grades – each grade (A, B, C, D, E) has a range of values that reflect each grade letter

Today

- Loop statements

Warm-up so far in this course

Loop is where it starts to get harder

I suggest you to spend more time on examples given in this and future lectures

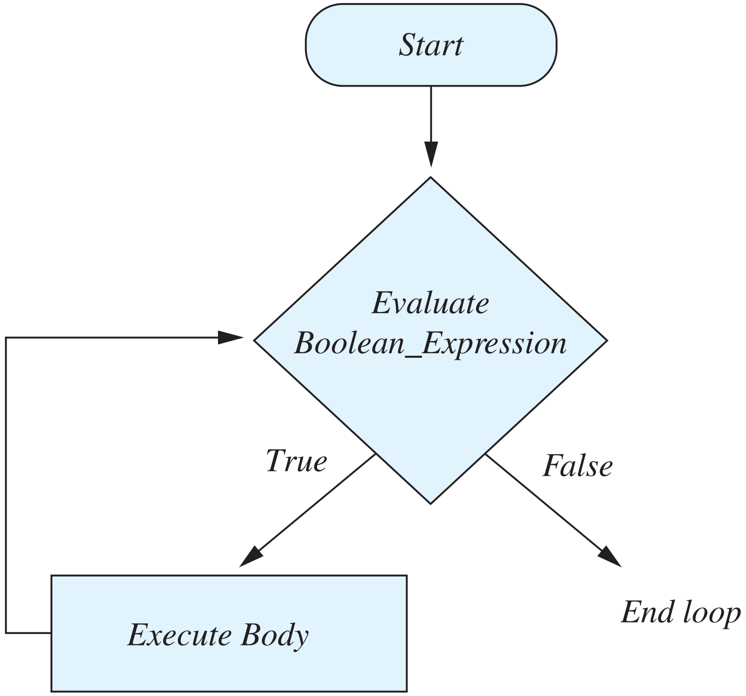
Flow of Control

- Alter the order in which a program’s statements are executed
- Typically, two kinds
  - Conditionals (if-else and switch)
    - Execute a set of statements by choosing among two or more paths
  - Loops
    - Repeat a group of instructions numerous times

Types of Loops

- while loop
  - Repeats its body while a boolean expression is true
- do while loop
  - Loop iterates at least ONCE
- for loop
  - Numeric computation changes by equal amount

The While Loop



- Syntax

  while (Boolean_Expression)

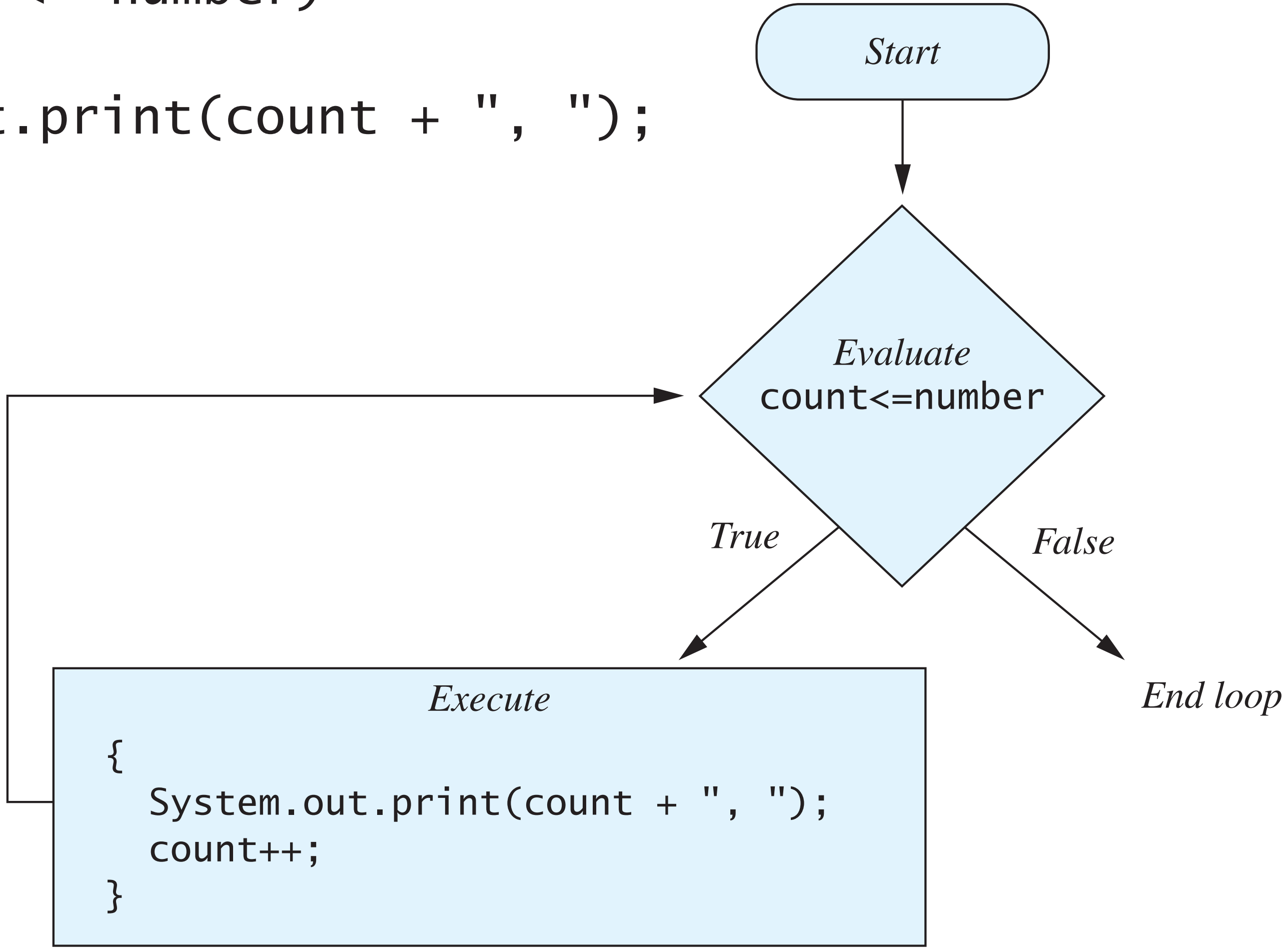
  Body

  - The Body may be either a simple statement or, a list of statements enclosed in braces {}

- A while statement repeats while a controlling boolean expression remains true
- The loop body typically contains an action that ultimately causes the controlling boolean expression to become false

An Example of a While Loop

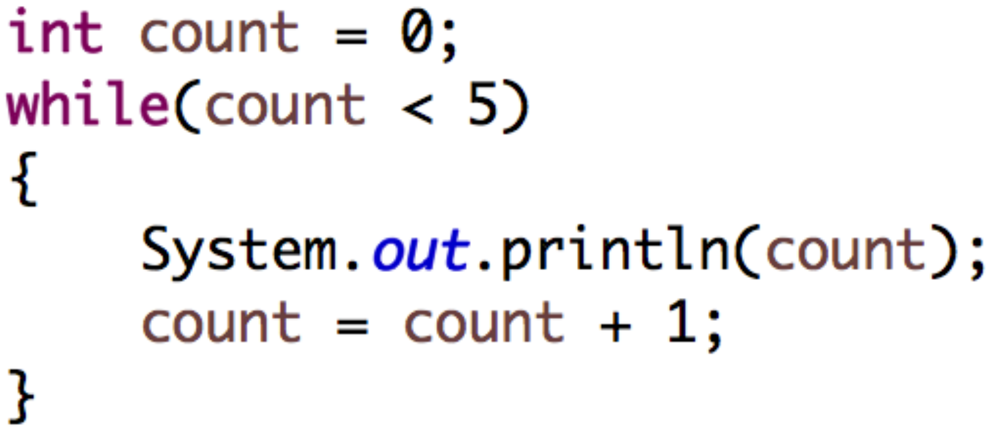
```
while (count <= number)
{
    System.out.print(count + ", ");
    count++;
}
```



```
{
  System.out.print(count + ", ");
  count++;
}
```

While in Practice

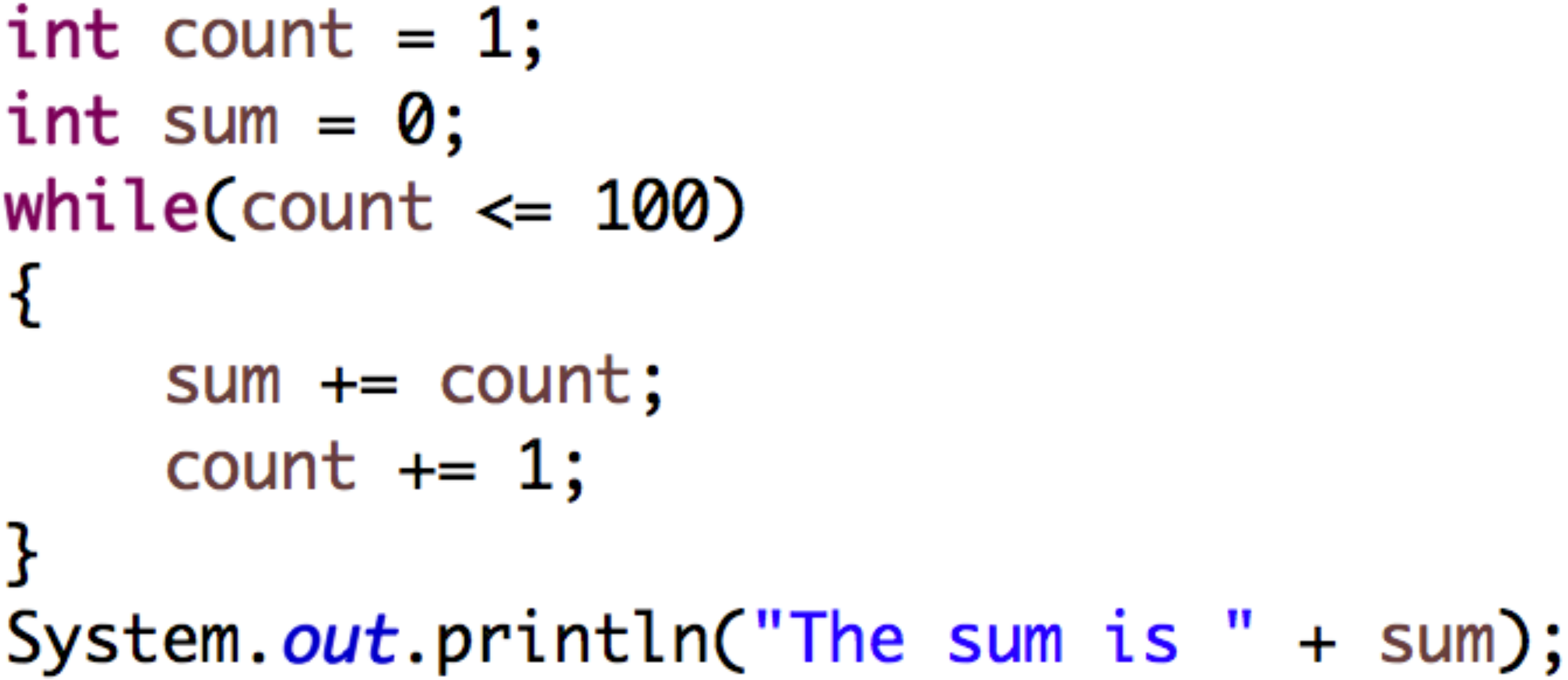
- What's the output?



```
int count = 0;
while(count < 5)
{
    System.out.println(count);
    count = count + 1;
}
```

  - Iteration 1: count = 0, < 5? true, print 0, +1
  - Iteration 2: count = 1, < 5? true, print 1, +1
  - …
  - Iteration 5: count = 4, < 5? true, print 4, +1
  - Iteration 6: count = 5, < 5? false, stop

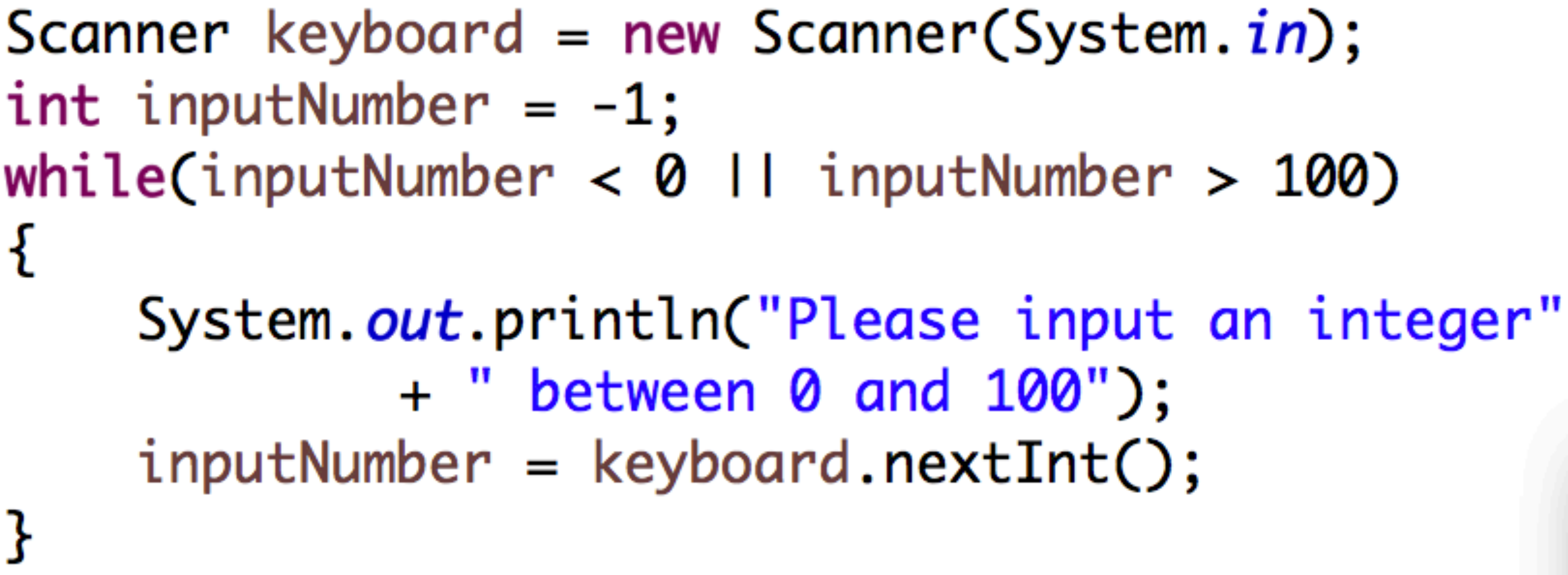
Calculate the Sum of 1…100



```
int count = 1;
int sum = 0;
while(count <= 100)
{
    sum += count;
    count += 1;
}
System.out.println("The sum is " + sum);
```

Input Checking

- Ask user to input an integer between 0 and 100, keep reading until we get the correct input



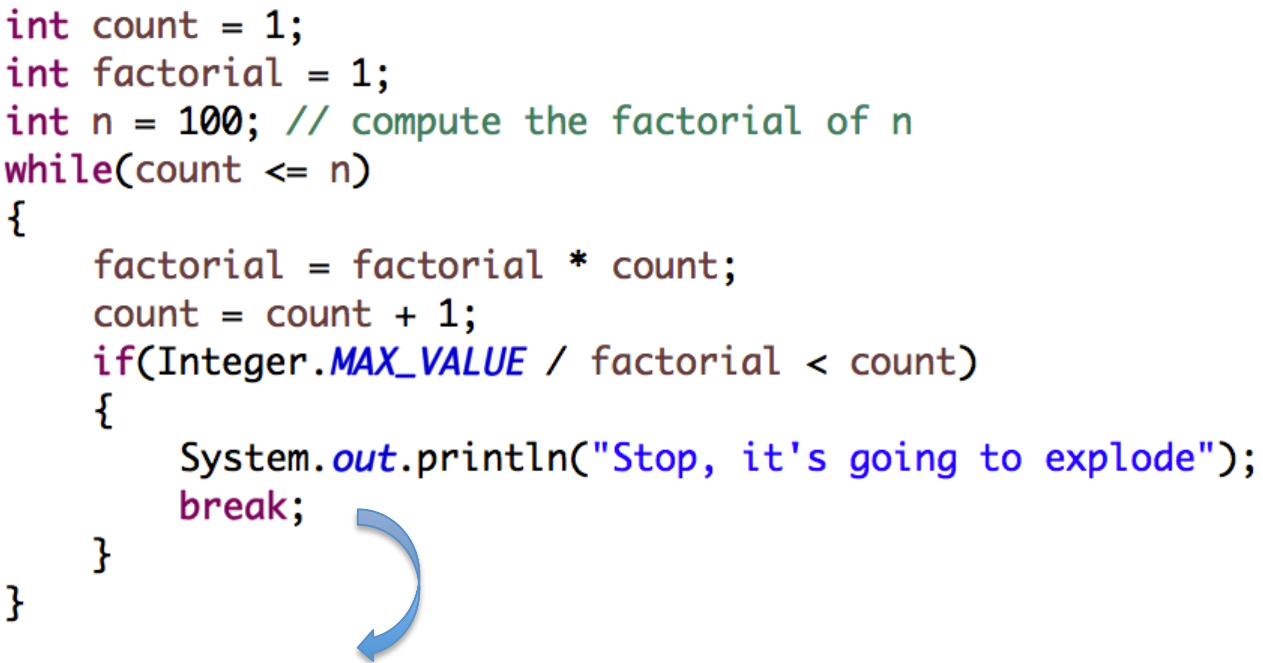
```java
Scanner keyboard = new Scanner(System.in);
int inputNumber = -1;
while(inputNumber < 0 || inputNumber > 100)
{
    System.out.println("Please input an integer"
            + " between 0 and 100");
    inputNumber = keyboard.nextInt();
}
```

Early Exit

break;

Exit a loop and continue to execute the statement after the loop

- Example: Compute factorial



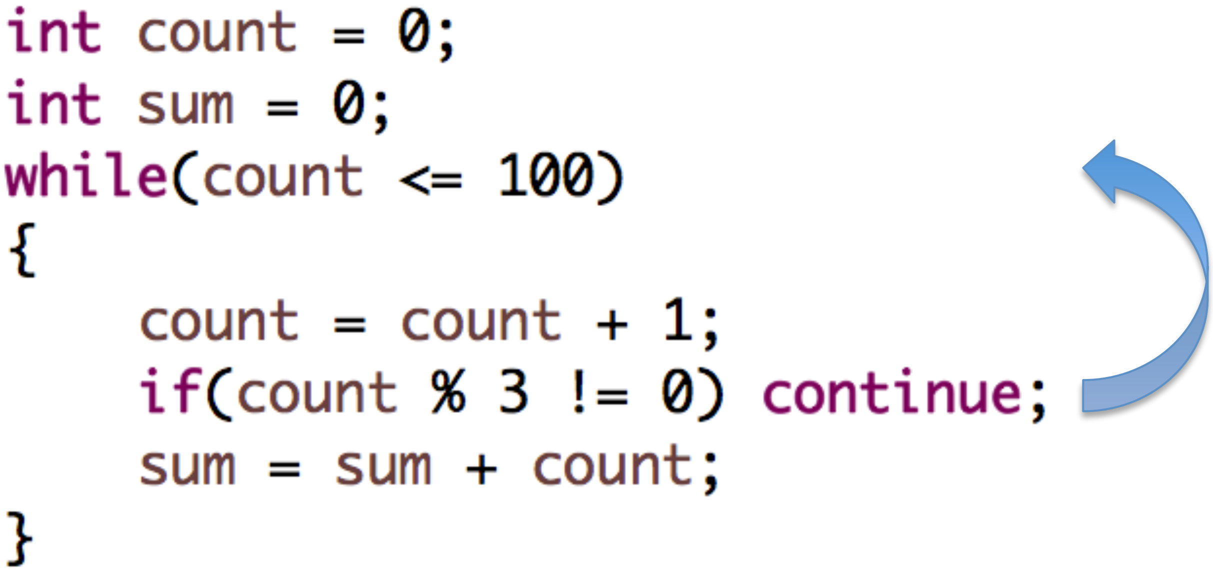
```
int count = 1;
int factorial = 1;
int n = 100; // compute the factorial of n
while(count <= n)
{
    factorial = factorial * count;
    count = count + 1;
    if(Integer.MAX_VALUE / factorial < count)
    {
        System.out.println("Stop, it's going to explode");
        break;
    }
}
```

Go To Next Iteration

continue;

Skip next part of a loop, and start the next iteration upon invocation

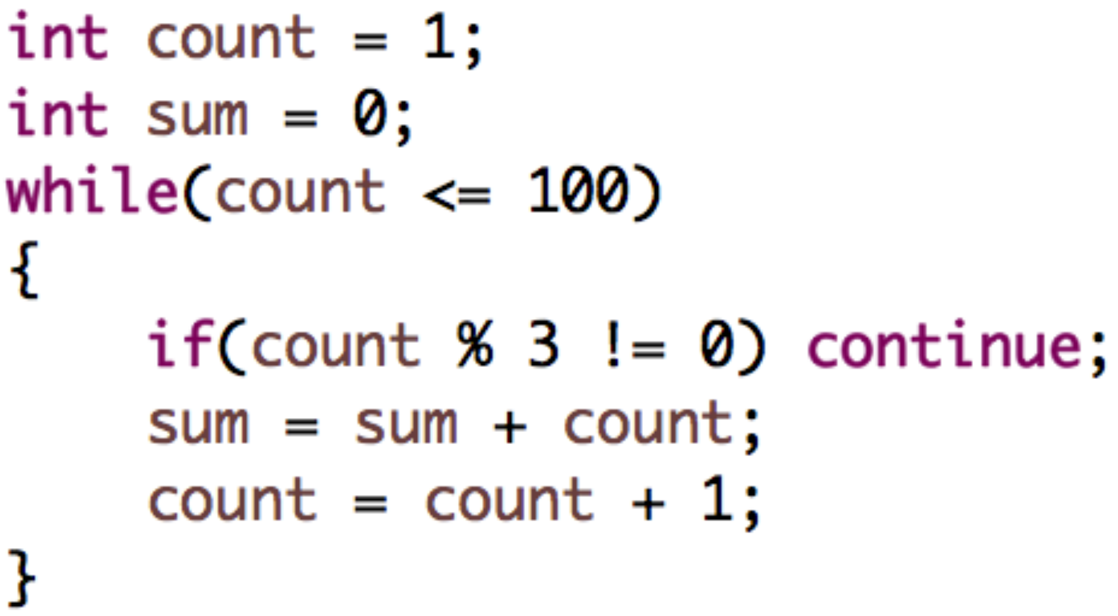
- Example: Calculate the sum of multiples of 3 within [1, 100]



```
int count = 0;
int sum = 0;
while(count <= 100)
{
    count = count + 1;
    if(count % 3 != 0) continue;
    sum = sum + count;
}
```

Compute the Sum of Multiples of 3

- What’s wrong with the following implementation?



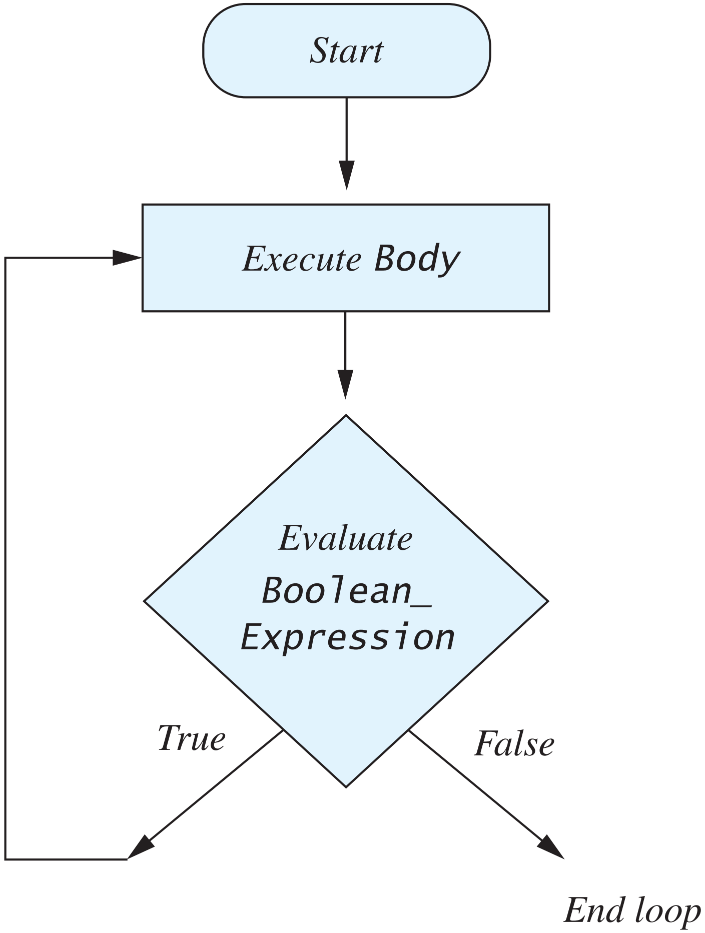
```
int count = 1;
int sum = 0;
while(count <= 100)
{
    if(count % 3 != 0) continue;
    sum = sum + count;
    count = count + 1;
}
```

The do-while Loop

- Similar to a while loop, except that the loop body is executed at least once
- Syntax

```
do
    Body
while (Boolean_Expression);
```

  - Don’t forget the semicolon



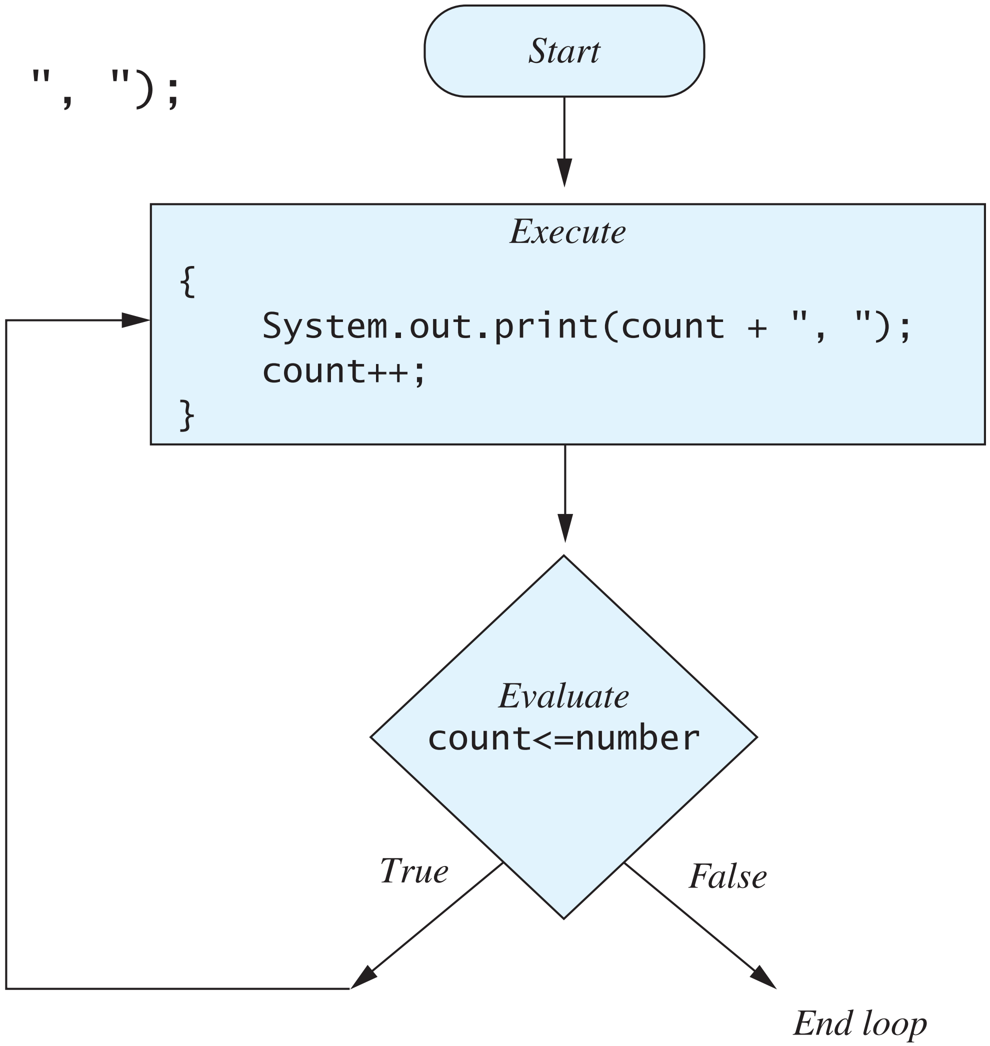
The do-while Loop

- First, the loop body is executed
- Then, the boolean expression is checked
  - As long as it is true, the loop is executed again
  - If it is false, the loop is exited
- Equivalent while statement

```
Statements
while (Boolean_Expression)
     Statements
```

An Example of the do-while Loop

```
do
{
   System.out.print(count + ", ");
   count++;
}  while (count <= number);
```



Loop Practice

- Write a while loop or a do-while loop that will compute the sum of the first n positive odd numbers. For example, if n is 5, you should compute 1 + 3 + 5 + 7 + 9.

Some short-forms

- Nested expression, used a lot in a loop
  - n = n + 1; → n++; or ++n;
  - n = n – 1; → n--; or --n;
  - n = n + m; → n += m;
  - n = n – m; → n -= m;

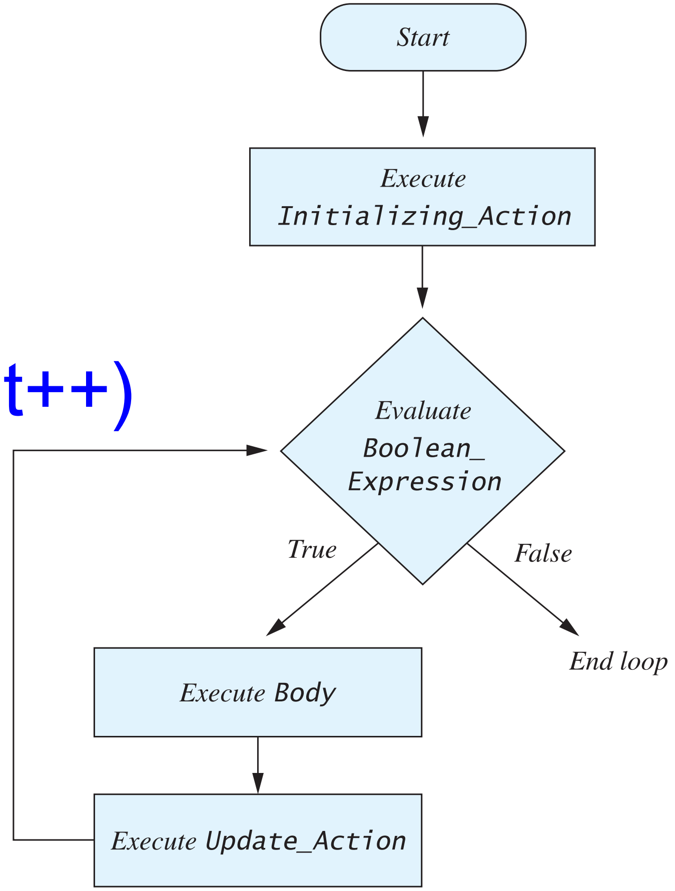
The for Loop

- Syntax

```
for (Initializing_Action; Boolean_Expression; Update_Action)
    Body
```

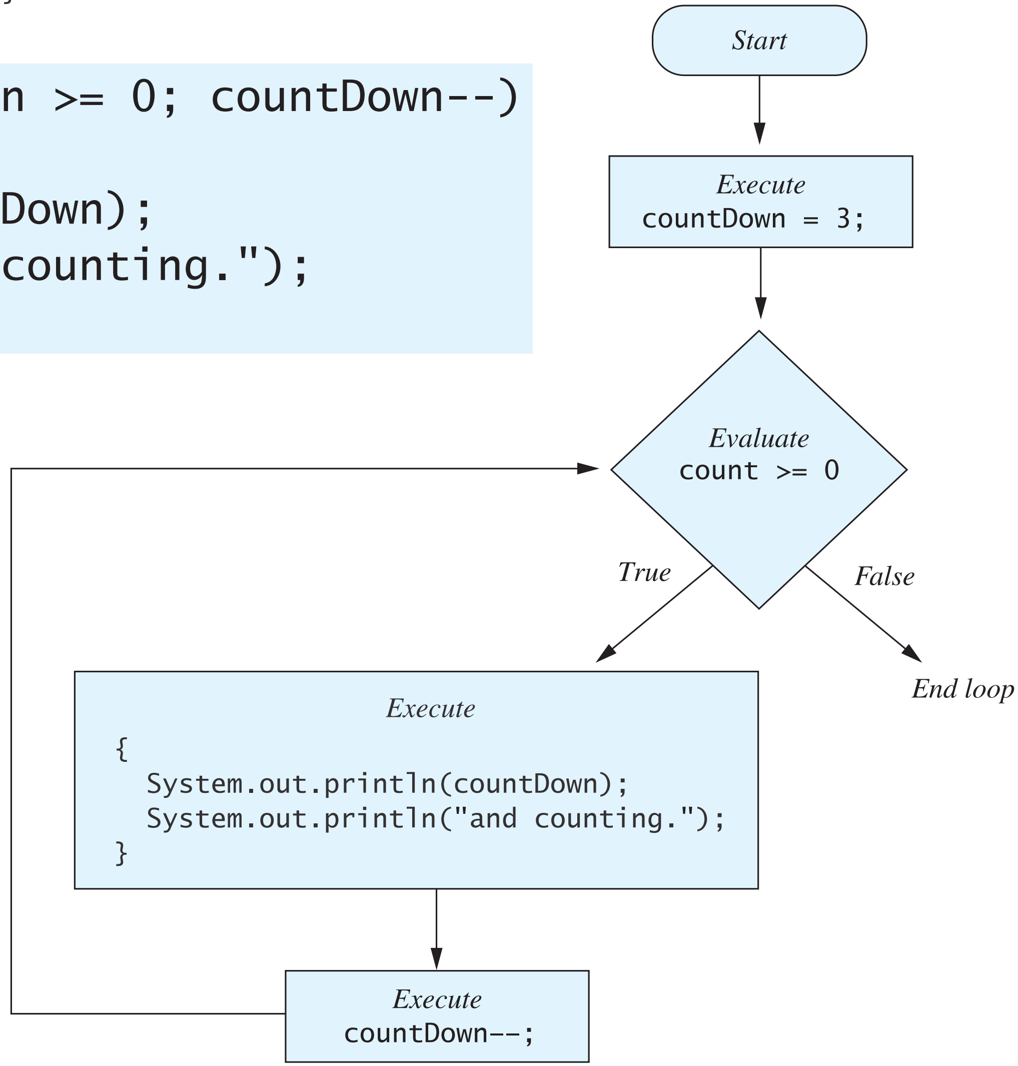
- Example

```
int count;
for (count = 1; count < 3; count++)
    System.out.println(count);
```

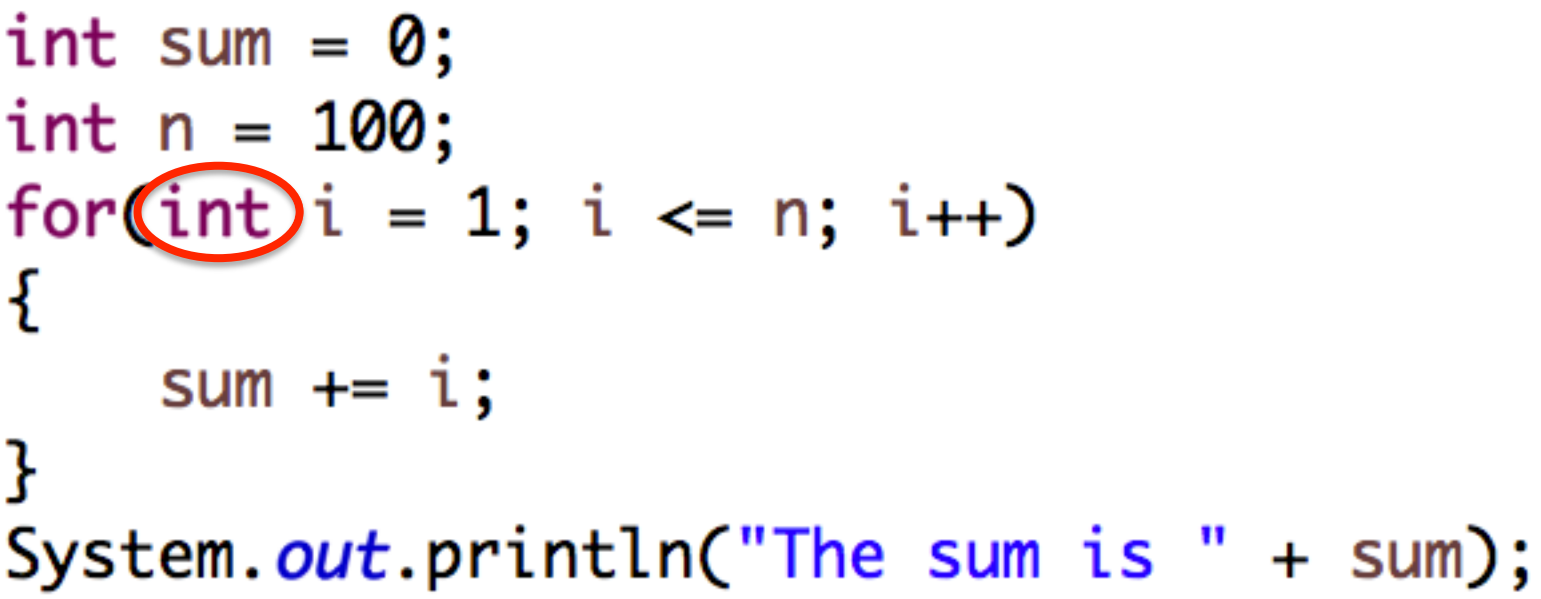


An Example of the for Loop

```
int countDown;
for (countDown = 3; countDown >= 0; countDown--)
{
    System.out.println(countDown);
    System.out.println("and counting.");
}
```



Another Example: The Sum of 1…n



```
int sum = 0;
int n = 100;
for(int i = 1; i <= n; i++)
{
    sum += i;
}
System.out.println("The sum is " + sum);
```

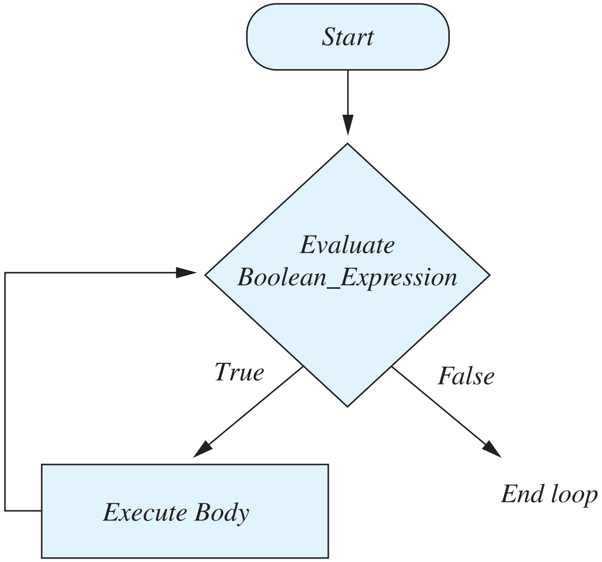
- Possible to declare variables within a for loop
- Note that variable i is local to the loop

Loop Practice

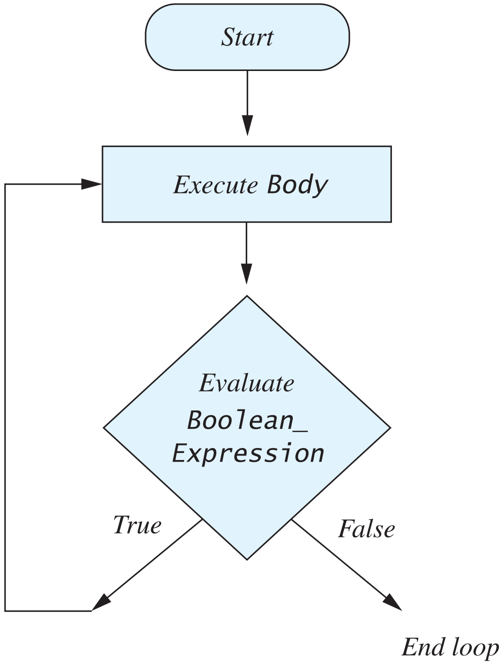
- Write a for loop that will compute the sum of the first n positive even numbers. For example, if n is 5, you should compute 2 + 4 + 6 + 8 + 10

Summary of Loops

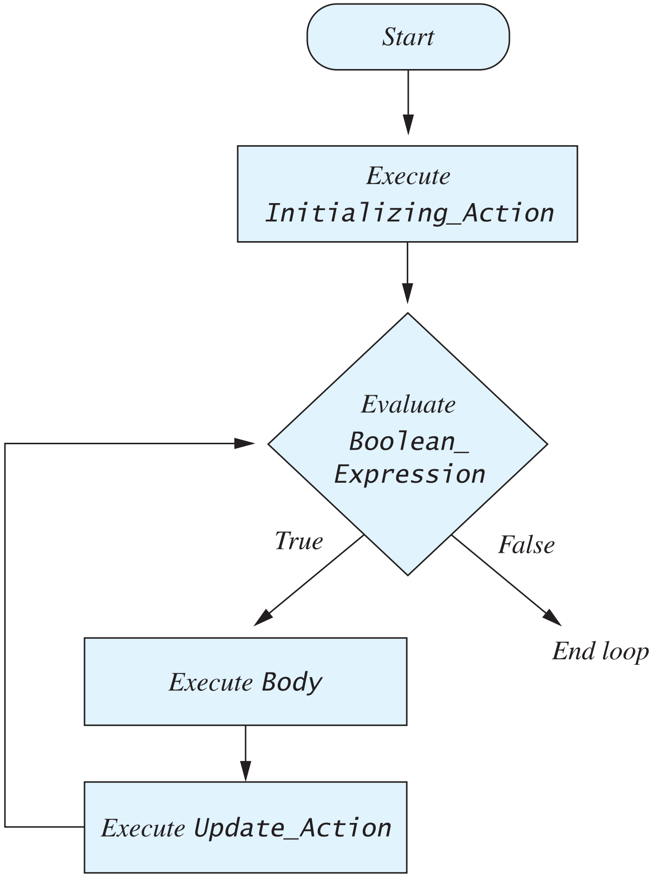
- Three types:



The **while** loop



The **do-while** loop



The **for** loop

Next Class

- More about Loops