CHAPTER 3

EXERCISE 1

3.10

**Comparison of if and while Statements**

**Similarities:**

1. **Both use a Boolean condition –**

The if and While Statements both evaluate a condition that must be trued and false.

1. **Both control the flow of execution –**

These statements determine whether certain code should execute based on a condition.

1. **Both can use blocks ({}) to execute multiple statements –**

If multiple statements need to be executed, they must be enclosed in curly braces {}.

|  |  |  |
| --- | --- | --- |
| Feature | If Statement (Single- Selection) | While Statement (Repetition) |
| Purpose | Used for conditional execution (one-time decision making). | Used for looping (repeating a block of code while a condition is true). |
| Execution | Executes the code block only once if the condition is true. | Repeats the code block as long as the condition remains true. |
| Iteration | Does not repeat execution. | Continuously checks the condition after each iteration and executes again if true. |
| Flow control | Used for making decisions (e. g. executing alternative code paths). | Used for repeating tasks, such as iterating through user input or processing data in loops |

**Differences:**

3.11

In java, when one integer is divided by another using the / operator, integer division occurs. This means:

* The fractional part (decimal portion) of the result is truncated (discarded)
* The result is always an integer, rounded toward zero.

3.12

In java, control statement can be combined in two main ways:

1. Nesting (Control Statements Within Control Statements)

Here one control statement can be placed inside another.

It allows for more complex decision-making or looping structures.

1. Sequential Execution (One Statement After Another)

Here control statements can be placed one after another in a sequence.

It allows execution to follow the order in which statements appear.

3.13

Calculating the sum of the First 100 Positive Integers

* Since we know the exact number of iterations (100), a for loop is the best choice.
* The sum can be computed by iterating from 1 to 100 and adding each number to a running total.

Calculating the Sum of an Arbitrary Number of Positive Integers

* Since we don’t know how many numbers the user will enter in advance, a while loop with sentinel-controlled repetition is appropriate.
* The loop continues until the user enters a sentinel value (e.g., -1 to stop).

3.14

In Pre-increment, the variable is incremented first and then its updated value is used in the expression.

In Post increment, the current value of the variable is used in the expression first and then the variable is incremented.

3.15

a.) Issue with if Statement

if (age >= 65) {

System.out .println(“Age is greater than or equal to 65”);

} else {

System.out.println(“Age is less than 65”);

}

b.) Issue with while loop

int x = 1, total = 0;

while (x <= 10) {

total += x;

++x;

}

c.) Issue with while loop missing braces

while (x <= 100) {

total += x;

++=;

}

d.) Issue with while loop (Infinite Loop)

while (y > 0) {

System.out.println(y);

--y;

}