Assignment 3

SECTION 1: Error-Driven Learning Assignment: Loop Errors

Snippet 1:

```
public class InfiniteForLoop {
    public static void main(String[] args) {
        for (int i = 0; i < 10; i--) {
            System.out.println(i);
        }
    }
}
// Error to investigate: Why does this loop run infinitely? How should the loop control variable be adjusted?
```

Ans:-

Error is that this loop runs infinitely because the condition always matches for for loop i.e (i<10).

It start form 0 and go till less then 10 and foe every I we decrement it. So condition always true and it always go in infinite loop.

Loop control variable are adjusted by doing i++ instead of decrement when we do increment so it gives o/p 1-10.

Snippet 2:

```
public class IncorrectWhileCondition {
    public static void main(String[] args) {
        int count = 5;
        while (count = 0) {
            System.out.println(count);
            count--;
        }
    }
}
// Error to investigate: Why does the loop not execute as expected? What is the issue with the condition in the
`while` loop?
```

Ans:-

error: incompatible types: int cannot be converted to boolean

```
    while (count = 0) {
                ^
```

1 error

Error is count =0 instad of it we have to write (count<=0) then it will print form 1 -5;

Ans

Snippet 3:

```
public class DoWhileIncorrectCondition {
   public static void main(String[] args) {
      int num = 0;
      do {
         System.out.println(num);
         num++;
      } while (num > 0);
```

```
   }
}
// Error to investigate: Why does the loop only execute once? What is wrong with the loop condition in the `do-while` loop?
```

Ans:-

The loop runs infinitely and print from 0-infinite because the while condition is that while (num>0)

Instead of that we have to write while(num<=10) so it print from 1-10 number.

Snippet 4:

```
public class OffByOneErrorForLoop {
   public static void main(String[] args) {
      for (int i = 1; i <= 10; i++) {
         System.out.println(i);
      }
      // Expected: 10 iterations with numbers 1 to 10
      // Actual: Prints numbers 1 to 10, but the task expected only 1 to 9
   }
}
// Error to investigate: What is the issue with the loop boundaries? How should the loop be adjusted to meet the expected output?
```

Ans:-

No error but the expected o/p is print from 1-9 but it print from 1-10 for this we can change the for loop in the condition of for loop we can remove =sign the it will be i<10 so it doesn't check for 10 and check till 9.

Snippet 5:

```
public class WrongInitializationForLoop {
   public static void main(String[] args) {
      for (int i = 10; i >= 0; i++) {
         System.out.println(i);
      }
   }
}
// Error to investigate: Why does this loop not print numbers in the expected order? What is the problem with the
initialization and update statements in the `for` loop?
```

Ans:-

The statement is start from 10 and goes till 0 and in snippet given is i>=0 so condition doesn't match and loop runs infinitely so instead of that we have to run it from 10 to 0 so condition is like i<=0 and we have to decrement the I for every iteration.

Snippet 6:

```
public class MisplacedForLoopBody {
   public static void main(String[] args) {
      for (int i = 0; i < 5; i++)
         System.out.println(i);
         System.out.println("Done");
   }
}
// Error to investigate: Why does "Done" print only once, outside the loop? How should the loop body be enclosed to
include all statements within the loop?
```

Ans:-

The code doesn't print done after printing every number because for loop see only one statement after coming in the for loop i.e it print 1234 first the print done .

Snippet 7:

```
public class UninitializedWhileLoop {
   public static void main(String[] args) {
      int count;

      while (count < 10) {
         System.out.println(count);
         count++;
      }
   }
}
// Error to investigate: Why does this code produce a compilation error? What needs to be done to initialize the loop
variable properly?
```

Ans

error: variable count might not have been initialized

```
      while (count < 10) {
            ^
```

1 error

There is error in the code that is count is not initialized so to compare while loop i.e count<10 we have to initialize the count to 0.

Snippet 8:

```
public class OffByOneDoWhileLoop {
   public static void main(String[] args) {
      int num = 1;
      do {
         System.out.println(num);
         num--;
      } while (num > 0);
   }
}
// Error to investigate: Why does this loop print unexpected numbers? What adjustments are needed to print the
numbers from 1 to 5?
```

Ans:-

Here the do while loop is it then we initialize num to 1 instead of that we have to initialized it to the 10 then it print 10-1.

Snippet 9:

```
public class InfiniteForLoopUpdate {
   public static void main(String[] args) {
      for (int i = 0; i < 5; i += 2) {
         System.out.println(i);
      }
   }
}
// Error to investigate: Why does the loop print unexpected results or run infinitely? How should the loop update
expression be corrected?
```

Ans:-

This loop not run infinitely because it start from 0 and gors till less then 5 so condition is correct and it increment the I by +2 si I will also increment by 2.

Snippet 10:

```
public class IncorrectWhileLoopControl {
   public static void main(String[] args) {
      int num = 10;
      while (num = 10) {
         System.out.println(num);
         num--;
      }
   }
}
// Error to investigate: Why does the loop execute indefinitely? What is wrong with the loop condition?
```

Ans

error: incompatible types: int cannot be converted to boolean

```
    while (num = 10) {
                 ^
```

1 error

In while there is condition but in this case there is assignment operator and compiler thinks it a Boolean type we have to convert so we have to put condition (num>=0) then it print 10-1.

Snippet 11:

```
public class IncorrectLoopUpdate {
    public static void main(String[] args) {
        int i = 0;
        while (i < 5) {
            System.out.println(i);
            i += 2; // Error: This may cause unexpected results in output
        }
    }
}
// Error to investigate: What will be the output of this loop? How should the loop variable be updated to achieve the
desired result?
```

Ans;

Output:-

0

2

4

Loop variable be updated as i++ instead of i+=2.

Snippet 12:

```
public class LoopVariableScope {
    public static void main(String[] args) {
        for (int i = 0; i < 5; i++) {
            int x = i * 2;
        }
        System.out.println(x); // Error: 'x' is not accessible here
    }
}
// Error to investigate: Why does the variable 'x' cause a compilation error? How does scope
```

Ans;

X is not accessible outside of for loop because the variable is initialize in for loop so it use in for loop outside of it there is no use of it.

SECTION 2: Guess the Output

Snippet 1:

```
public class NestedLoopOutput {
    public static void main(String[] args) {
        for (int i = 1; i <= 3; i++) {
            for (int j = 1; j <= 2; j++) {
                System.out.print(i + " " + j + " ");
            }
            System.out.println();
```

[redacted]

```
        }
    }
}
// Guess the output of this nested loop.
```

OUTPUT

1 1 1 2

2 1 2 2

3 1 3 2

Snippet 2:

```
public class DecrementingLoop {
   public static void main(String[] args) {
      int total = 0;
      for (int i = 5; i > 0; i--) {
         total += i;
         if (i == 3) continue;
         total -= 1;
      }
      System.out.println(total);
   }
}
// Guess the output of this loop.
```

Output:

11

Snippet 3:

```
public class WhileLoopBreak {
   public static void main(String[] args) {
      int count = 0;
      while (count < 5) {
         System.out.print(count + " ");
         count++;
         if (count == 3) break;
      }
      System.out.println(count);
   }
}
// Guess the output of this while loop.
```

OUTPUT:-

0 1 2 3

Snippet 4:

```
public class DoWhileLoop {
    public static void main(String[] args) {
        int i = 1;
        do {
            System.out.print(i + " ");
            i++;
        } while (i < 5);
        System.out.println(i);
    }
}
// Guess the output of this do-while loop.
```

OUTPUT:-

1 2 3 4 5

Snippet 5:

```
public class ConditionalLoopOutput {
    public static void main(String[] args) {
        int num = 1;
        for (int i = 1; i <= 4; i++) {
            if (i % 2 == 0) {
                num += i;
            } else {
                num -= i;
            }
        }
        System.out.println(num);
    }
}
// Guess the output of this loop.
```

OUTPUT:-

3

Snippet 6:

```
public class IncrementDecrement {
    public static void main(String[] args) {
        int x = 5;
        int y = ++x - x-- + --x + x++;
        System.out.println(y);
    }
}
// Guess the output of this code snippet.
```

OUTPUT:-

8

Snippet 7:

```
public class NestedIncrement {
    public static void main(String[] args) {
        int a = 10;
        int b = 5;
        int result = ++a * b-- - --a + b++;
        System.out.println(result);
    }
}
// Guess the output of this code snippet.
```

OUTPUT:-

49

Snippet 8:

```
public class LoopIncrement {
    public static void main(String[] args) {
        int count = 0;
        for (int i = 0; i < 4; i++) {
            count += i++ - ++i;
        }
        System.out.println(count);
    }
}
// Guess the output of this code snippet.
```

OUTPUT:-

-4