Assignment no.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 the loop not execute as expected? What is the issue with the condition in the

`while` loop?

ans:-

the for loop runs infinitely becaus of loop control variable.the loop initializes 'i' to '0'and the condition is 'i<10' that's why the code runs infinitely.

to correct the code we need to adjust the loop control variable such as:

```
for(i=0;i<10;i--)
```

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:-

= operator is an assignment operator, not a comparison operator. This means that the expression count = 0 is an assignment operation, not a condition check

(in Boolean value 0 means false).

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 `dowhile` loop?

ans:-

the code execute infinitely.int num = 0; initializes num to 0.System.out.println(num); prints 0.

num++; increments num to 1.

The loop condition while (num > 0) is then checked.

num is now 1, which is greater than 0, the condition evaluates to true.

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:-

given code prints the value from 1 to 10 bcz we initialize i=1 and the condition is i<=10 the condition gets executed until the i=10

if we want to print the values from 1 to 9 then we need to change the condition as : i<10

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:-

we initialize i=10.and the update statement i++ increase the value of i by 1 in each iteration.Since i is incrementing, the condition i >= 0 will always be true once i is greater than 0, causing an infinite loop.

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 loop prints numbers from 0 to 4, as expected.

After the loop completes (when i reaches 5), the program moves on to the next statement, which is System.out.println("Done");.

Since this statement is not part of the loop body (due to the lack of braces {}), it executes only once, after the loop has finished.

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:-

The variable count is declared but not initialized before being used in the while loop condition. In Java, a variable must be initialized before it is used.

In the while loop, count is used in the condition count < 10. Since count was not initialized, the compiler does not know what value count holds at this point.

To fix the error, you need to initialize the variable count before the while loop. This ensures that count has a known value when the loop begins.(count = 1)

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:-

the loop print unexpectec numbers because it is set to decrement the num variable instead of incrementing it

To print numbers from 1 to 5:-

Initialize num to 1

Change the loop body to increment num instead of decrementing it.

Update the loop condition to ensure it continues as long as num is less than or equal to 5.(num<=5)

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:-

the code does not run infinitely. it print the vale as(0,2,4)

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:-

runs indefinitely due to a mistake in the loop condition.

while (num = 10) contains an assignment num = 10 instead of a comparison. This means that the condition is always assigning 10 to num, not checking if num is equal to 10.

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:-

the output of this code is (0,4,6)

int i = 0; initializes the loop variable i to 0

while (i < 5) continues the loop as long as i is less than 5.

System.out.println(i); prints the current value of i.

i += 2; increments i by 2 each iteration.

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:-

The variable x is declared inside the for loop block: for (int i = 0; i < 5; i++) { int x = i * 2; }.

This means x is local to the for loop block.

variables declared within a block (enclosed by {}) are accessible only within that block. Once the block ends, the variable goes out of scope.

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();
 }
 }}
// Guess the output of this nested loop.
```

Ans:-

I=1	j=1	1	1	
	J=2	1	2	print – 1 1 1 2
I=2	j=1	2	1	
	J=2	2	2	print - 2 1 2 2
I=3	j=1	3	1	
	J=2	3	2	print – 3 1 3 2

<u>Output:-</u>

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.
```

Ans:

total=0

1] i=5

i>0 5>0 true

total+=I total = total+I

= 0+5=5

If i==3 continue (false)

total -= 1 total= total – 1

= 5-1 = 4

2] I-- = i=4

i>0 4>0 true

total+=I total = total+I

= 4+4=8

If i==3 continue (false)

total -= 1 total= total – 1

= 8-1=7

Output:- 11

3] i-- = i=3

i>0 3>0 true

total+=I total = total+i

= 7+3=10

If i==3 continue (true)

Skip total -= 1

4] I-- = i=2

i>0 2>0 true

total+=I total = total+i

= 10+2=12

total -= 1 total= total – 1

= 12-1=11

5] I-- = i=1

i>0 1>0 true

total+=I total = total+i

= 11+1=12

total -= 1 total= total – 1

= 12-1=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.
```

Ans:

Count=0

Condition : 0<5 true

Print – 0

Count++ = 1

Condition:1<5 true

Print – 1

Count++ = 2

Condition: 2<5 true

Print- 2

Count++ = 3

Condition: 3<5 true

Print-3

Count == 3 true break

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.

Ans:-

I=1

Print -1

I++ = 2

Condition:- (2<5) true

Print- 2

I++ =3

Condition:- (3<5) true

Print -3

I++ = 4

Condition:- (4<5) true

Print-4

I++ =5

Condition:- (5<5) false

Statement Out side do while loop print 5 bcz the value of i is declared before loop and updated value of I =5

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
```

Ans:-

Num=1

I=1

1] i<=4 1<=4 true

If(1%2==0) false

Num-=I num=num-I

1-1=0

Num=0

2] i=2 2<=4 true if

If(2%2==0) true

Num+=I num=num+i

0+2 =2

Num=2

3] i=3 3<=4 true

If(3%2==0) false

Num-=I num=num-I

2-3=-1

Num = -1

4] i=4 4<=4 true

If(4%2==0) true

Num+=I num=num+i

-1+4=3

Num = 3

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.
```

Ans:-

X=5

++x=6

x--= 6 (post decrement)so use the value first then decrement

--x=5-1=4

X++=4(post increment) so use the value first then increment

6-6+4+4=8

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
```

Ans:-

```
 ++a = 11

 b-- = 5

 --a = 10
b++ = 4
```

11 * 5 - 10 + 4 = 49

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.

Ans:-

Count=0

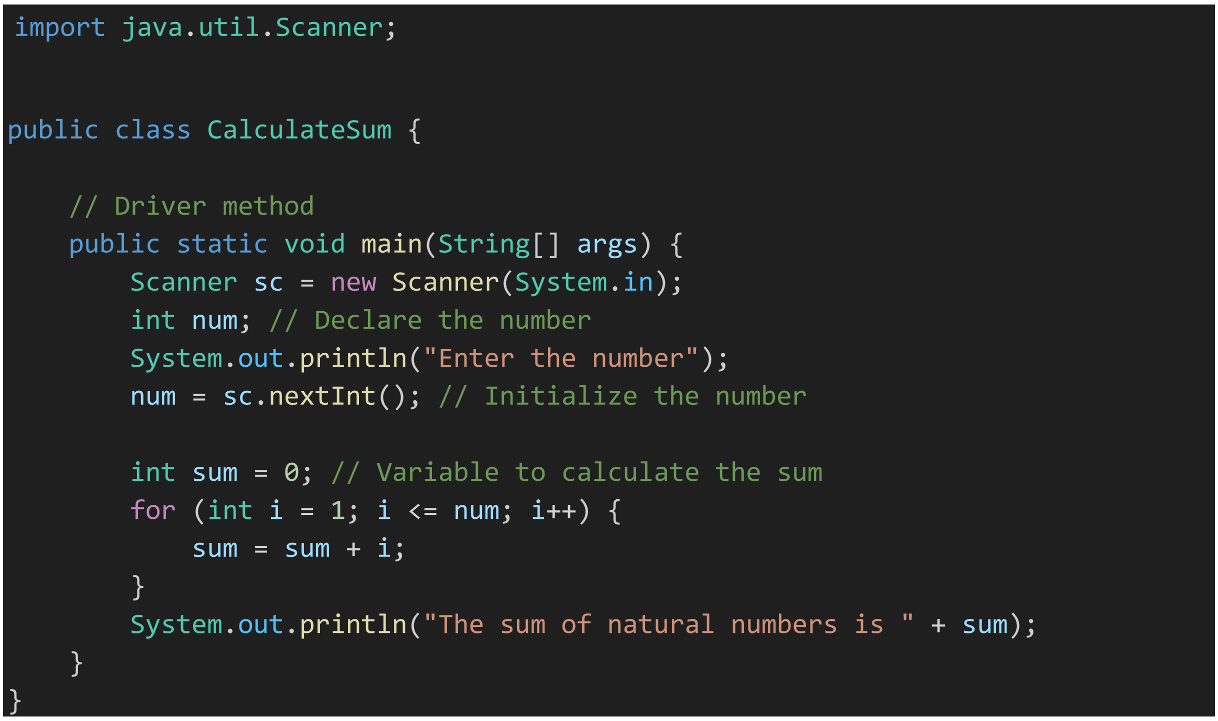
1] I=0;	2]i=1	3]i=2	4]i=3
(0<4) true	(1<4) true	(2<4)true	(3<4) true
Count=count+0-2	Count=-2+1-3	count=-2+2-3	count=-3+3-4
=-2	=-2	=-3	=-4

Output:-

-4

SECTION 3:

1. Write a program to calculate the sum of the first 50 natural numbers.



```java
 import java.util.Scanner;

public class CalculateSum {

    // Driver method
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int num; // Declare the number
        System.out.println("Enter the number");
        num = sc.nextInt(); // Initialize the number

        int sum = 0; // Variable to calculate the sum
        for (int i = 1; i <= num; i++) {
            sum = sum + i;
        }
        System.out.println("The sum of natural numbers is " + sum);
    }
}
```

2.Write a program to compute the factorial of the number 10.

```
public class Factorial {
    public static void main(String[] args) {
        int num = 10;
        long factorial = 1;
        for (int i = 1; i <= num; i++) {
            factorial *= i;
        }
        System.out.println("Factorial of " + num + " is " + factorial);
    }
}
```

3.Write a program to print all multiples of 7 between 1 and 100.

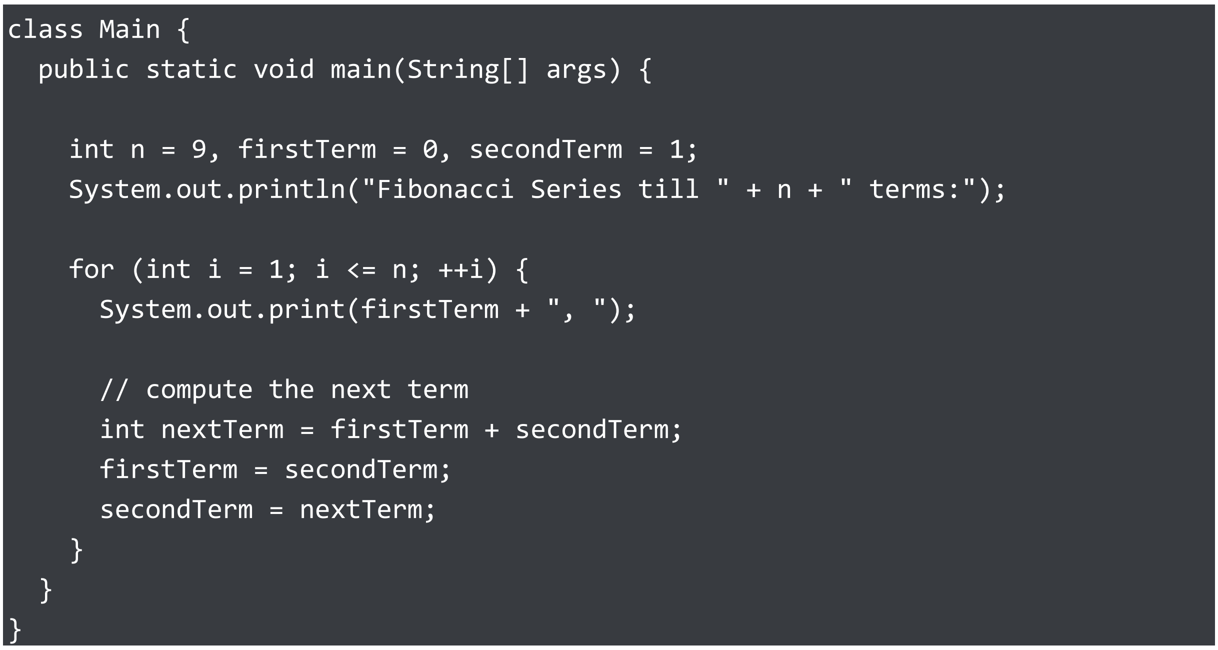
```

public class multiplication {
    public static void main(String[] args) {
        for (int i = 7; i <= 100; i += 7) {
            System.out.println(i + " ");
        }
    }
}
```

4. Write a program to reverse the digits of the number 1234. The output should be 4321

```
import java.util.*;
class Reverse {
    public static void main(String[] args) {
        System.out.println("Enter the number: ");
        Scanner sc = new Scanner(System.in);
        int num = sc.nextInt();
        int reverse = 0;
        while (num != 0) {
            int remainder = num % 10;
            reverse = reverse * 10 + remainder;
            num = num / 10;
        }
        System.out.println("reverse num is: " + reverse);
    }
}
```

5. Write a program to print the Fibonacci sequence up to the number 21.



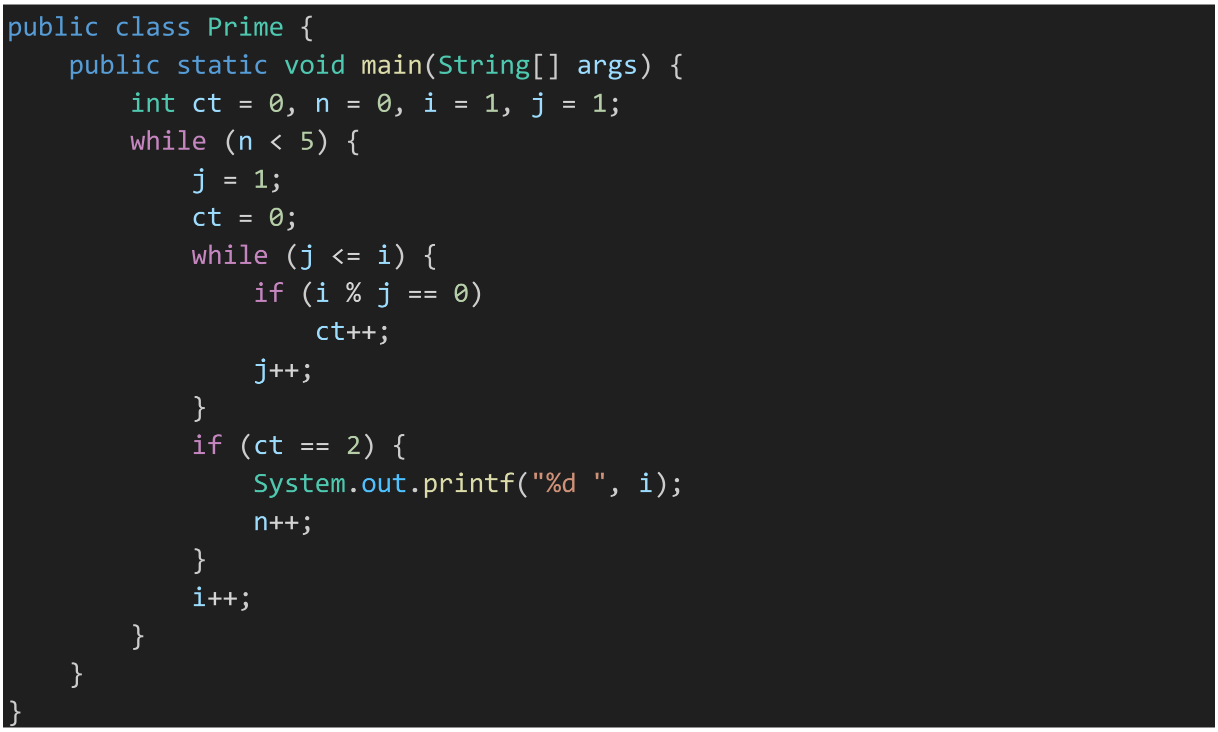
```
class Main {
  public static void main(String[] args) {

    int n = 9, firstTerm = 0, secondTerm = 1;
    System.out.println("Fibonacci Series till " + n + " terms:");

    for (int i = 1; i <= n; ++i) {
      System.out.print(firstTerm + ", ");

      // compute the next term
      int nextTerm = firstTerm + secondTerm;
      firstTerm = secondTerm;
      secondTerm = nextTerm;
    }
  }
}
```

6. Write a program to find and print the first 5 prime numbers.



```
public class Prime {
    public static void main(String[] args) {
        int ct = 0, n = 0, i = 1, j = 1;
        while (n < 5) {
            j = 1;
            ct = 0;
            while (j <= i) {
                if (i % j == 0)
                    ct++;
                j++;
            }
            if (ct == 2) {
                System.out.printf("%d ", i);
                n++;
            }
            i++;
        }
    }
}
```

7. Write a program to calculate the sum of the digits of the number 9876. The output should be 30 (9 + 8 + 7 + 6).

```java
import java.util.Scanner;

public class SumOfDigits {
    public static void main(String args[]) {
        int number, digit, sum = 0;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the number: ");
        number = sc.nextInt();
        while (number > 0) {

            digit = number % 10;

            sum = sum + digit;

            number = number / 10;
        }

        System.out.println("Sum of Digits: " + sum);
    }
}
```

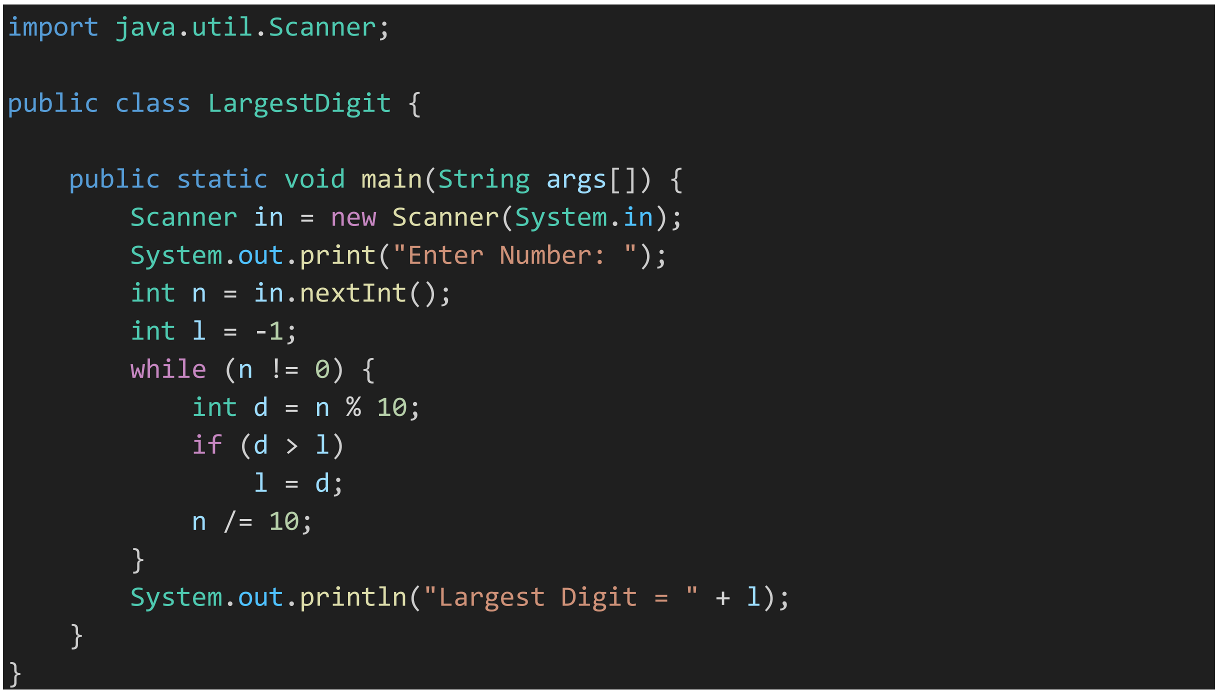
8. Write a program to count down from 10 to 0, printing each number.

```java
public class Countdown {

    public static void main(String[] args) {
        //
        for (int i = 10; i >= 0; i--) {
            System.out.println("i = " + i);
        }

    }
}
```

9. Write a program to find and print the largest digit in the number 4825

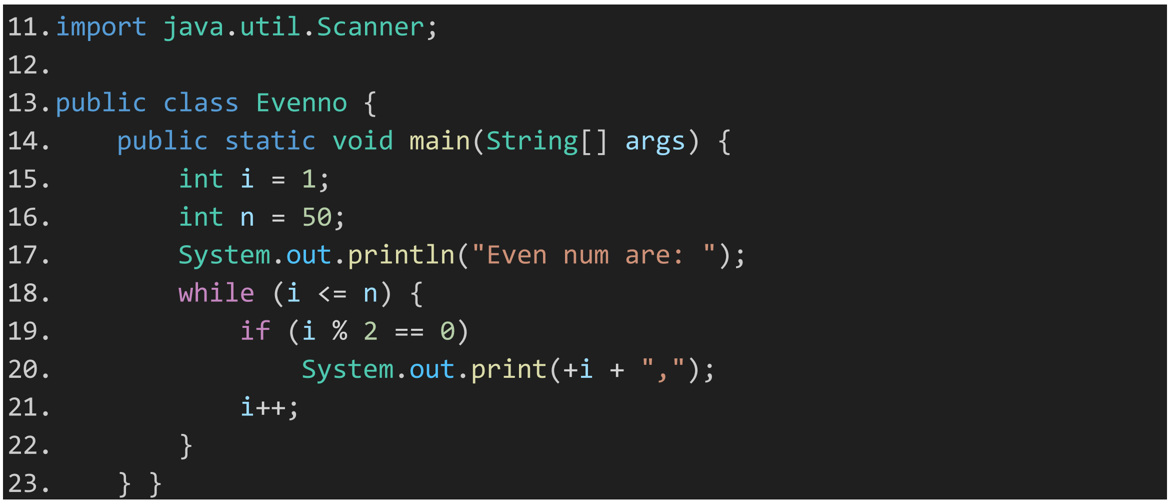


```java
import java.util.Scanner;

public class LargestDigit {

    public static void main(String args[]) {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter Number: ");
        int n = in.nextInt();
        int l = -1;
        while (n != 0) {
            int d = n % 10;
            if (d > l)
                l = d;
            n /= 10;
        }
        System.out.println("Largest Digit = " + l);
    }
}
```

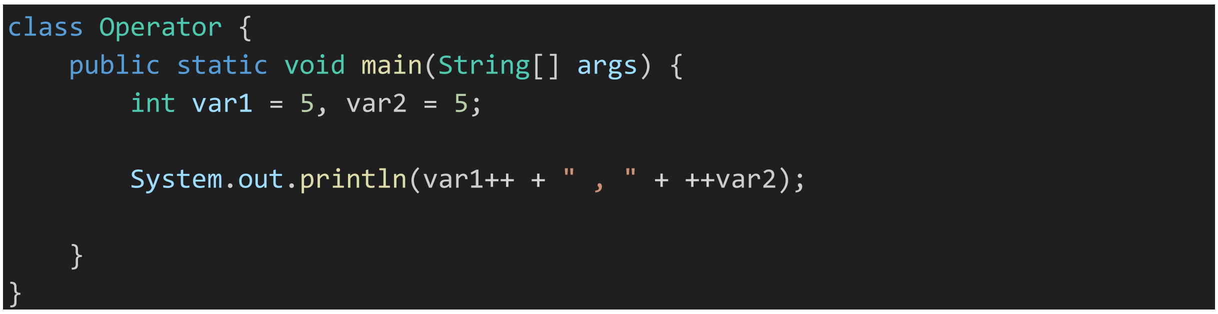
10. Write a program to print all even numbers between 1 and 50.



```java
import java.util.Scanner;

public class Evenno {
    public static void main(String[] args) {
        int i = 1;
        int n = 50;
        System.out.println("Even num are: ");
        while (i <= n) {
            if (i % 2 == 0)
                System.out.print(+i + ",");
            i++;
        }
    } }
```

11. Write a Java program to demonstrate the use of both pre-increment and post-decrement operators in a single expression



```java
class Operator {
    public static void main(String[] args) {
        int var1 = 5, var2 = 5;

        System.out.println(var1++ + " , " + ++var2);

    }
}
```

12. Write a program to draw the following pattern:

```
*****
*****
*****
*****
*****
```

```
public class Star1 {
    public static void main(String[] args) {
        int a, b;
        for (int i = 0; i < 5; i++) {
            for (int j = 0; j < 5; j++) {
                System.out.print("*");

            }
            System.out.println();

        }
    }

}
```

13. Write a program to print the following pattern:

1

2*2

3*3*3

4*4*4*4

5*5*5*5*5

5*5*5*5*5

4*4*4*4

3*3*3

2*2

```
public class Star9 {
    public static void main() {
        int maxnum = 5;
        for (int i = 1; i <= maxnum; i++) {
            for (int j = 1; j <= i; j++) {
                if (j > 1)
                    System.out.print("*");
                System.out.print(i);
            }
            System.out.println();
        }
        for (int i = maxnum; i > 1; i--) {
            for (int j = 1; j <= i; j++) {
                if (j > 1)
                    System.out.print("*");
                System.out.print(i);
            }
            System.out.println();
        }
    }
}
```

14. Write a program to print the following pattern:

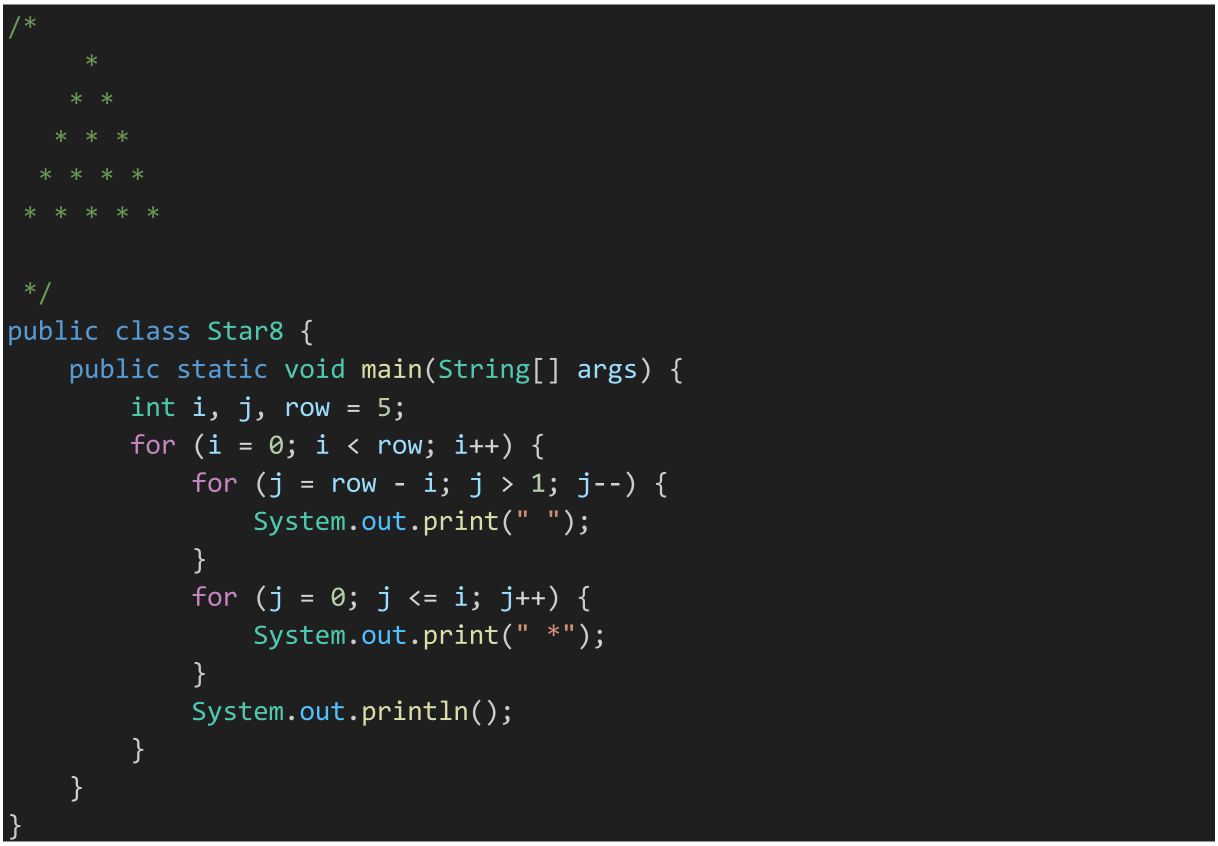
```
*
**
***
*****
*******
*********

public class Star12 {
    public static void main(String[] args) {

        for (int i = 1; i <= 9; i++) {
            if (i == 2 || i % 2 != 0) {
                for (int j = 1; j <= i; j++) {
                    System.out.print("*");
                }
                System.out.println();
            }
        }

    }
}
```

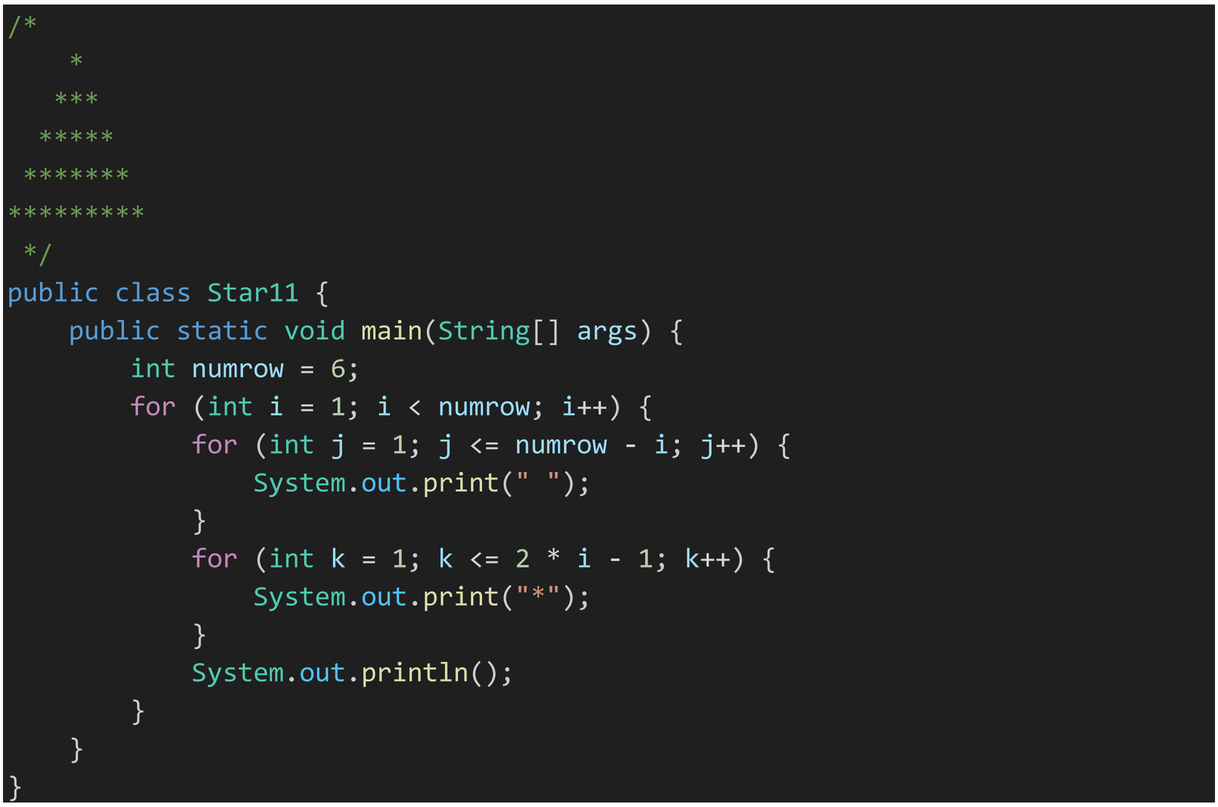
15. Write a program to print the following pattern:



```
/*
     *
    * *
   * * *
  * * * *
 * * * * *

 */
public class Star8 {
    public static void main(String[] args) {
        int i, j, row = 5;
        for (i = 0; i < row; i++) {
            for (j = row - i; j > 1; j--) {
                System.out.print(" ");
            }
            for (j = 0; j <= i; j++) {
                System.out.print(" *");
            }
            System.out.println();
        }
    }
}
```

16. Write a program to print the following pattern:



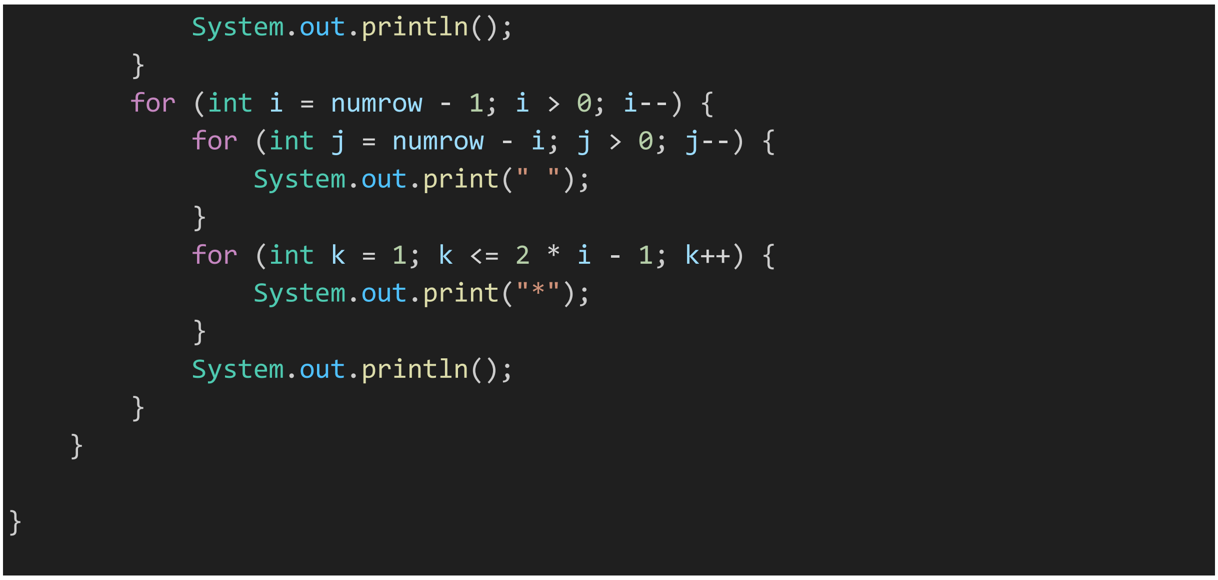
```
/*
    *
   ***
  *****
 *******
*********
 */
public class Star11 {
    public static void main(String[] args) {
        int numrow = 6;
        for (int i = 1; i < numrow; i++) {
            for (int j = 1; j <= numrow - i; j++) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

17. Write a program to print the following pattern:

```
/*
* * * * *
 * * * *
  * * *
   * *
    *
 */
public class Star13 {
    public static void main(String args[]) {
        int numrow = 5;
        for (int i = 0; i < numrow; i++) {
            for (int j = 0; j < i; j++) {
                System.out.print(" ");
            }
            for (int k = numrow - i; k > 0; k--) {
                System.out.print("* ");
            }
            System.out.println();
        }
    }
}
```

18. Write a program to print the following pattern:

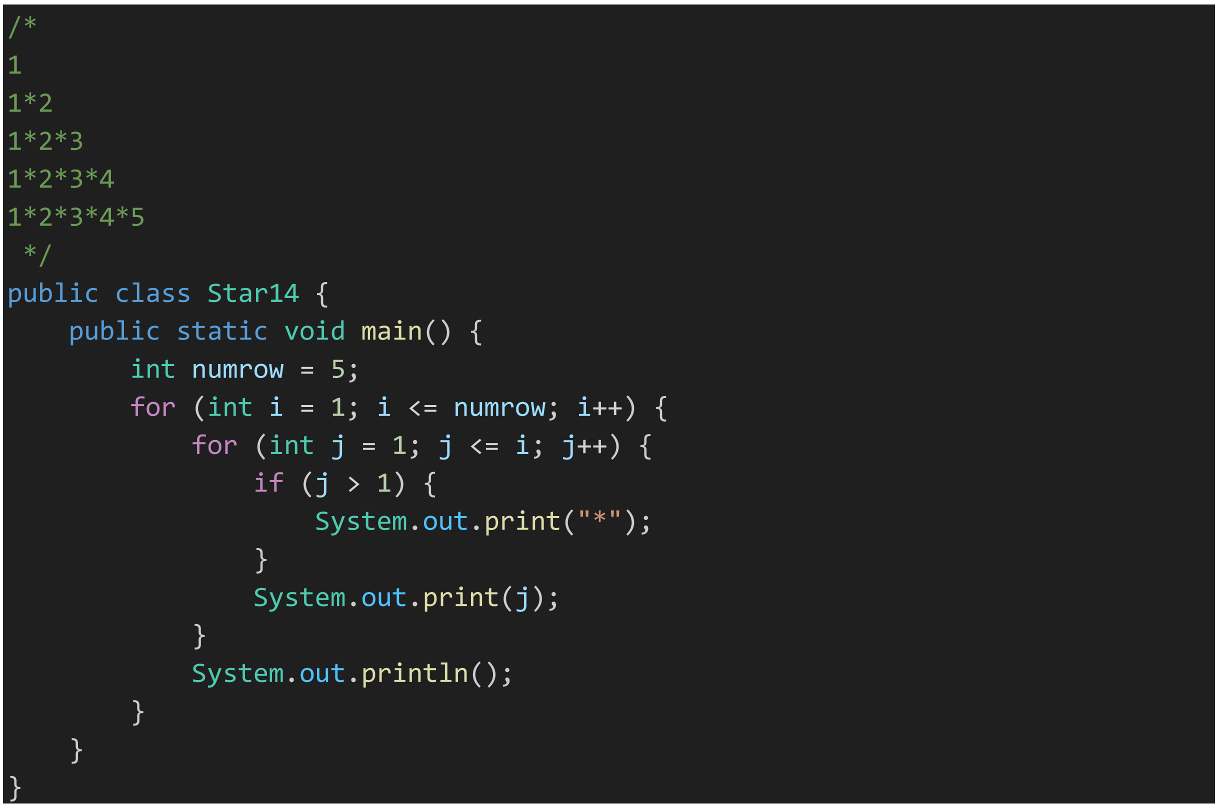
```
/*
   *
  ***
 *****
*******
 *****
  ***
   *
 */
public class Star17 {
    public static void main(String[] args) {
        int numrow = 4;
        for (int i = 1; i <= numrow; i++) {
            for (int j = numrow - i; j > 0; j--) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
```



```java
            System.out.println();
        }
        for (int i = numrow - 1; i > 0; i--) {
            for (int j = numrow - i; j > 0; j--) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }

}
```

19. Write a program to print the following pattern:



```java
/*
1
1*2
1*2*3
1*2*3*4
1*2*3*4*5
 */
public class Star14 {
    public static void main() {
        int numrow = 5;
        for (int i = 1; i <= numrow; i++) {
            for (int j = 1; j <= i; j++) {
                if (j > 1) {
                    System.out.print("*");
                }
                System.out.print(j);
            }
            System.out.println();
        }
    }
}
```

20. Write a program to print the following pattern:

```
/*
5
5*4
5*4*3
5*4*3*2
5*4*3*2*1
 */
public class Star16 {
    public static void main(String[] args) {
        int num = 5;
        for (int i = 0; i < num; i++) {
            for (int j = num; j > num - i - 1; j--) {
                if (j < num) {
                    System.out.print("*");
                }
                System.out.print(j);
            }
            System.out.println();
        }
    }
}
```

21. Write a program to print the following pattern:

```
/*
1
1*3
1*3*5
1*3*5*7
1*3*5*7*9
 */
public class Star15 {
    public static void main(String[] args) {
        int numrow = 5;
        for (int i = 1; i <= numrow; i++) {
            for (int j = 1; j <= i; j++) {
                if (j > 1) {
                    System.out.print("*");
                }
                System.out.print(2 * j - 1);
            }
            System.out.println();
        }
    }
}
```

22. Write a program to print the following pattern:

```
*********
 *******
  *****
   ***
    *
   ***
  *****
 *******
*********
public class Star10 {
    public static void main(String[] args) {
        int m = 9;
        int mrow = (m + 1) / 2;
        for (int i = mrow; i >= 1; i--) {
            for (int j = mrow - i; j > 0; j--) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
            System.out.println();
        }

        for (int i = 2; i <= mrow; i++) {
            for (int j = mrow - i; j > 0; j--) {
                System.out.print(" ");
            }
            for (int k = 1; k <= 2 * i - 1; k++) {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

23 Write a program to print the following pattern:

11111

22222

33333

44444

55555

```java
public class Star3 {
    public static void main(String[] args) {
        for (int i = 1; i <= 5; i++) {
            for (int j = 1; j <= 5; j++) {
                System.out.print(+i);
            }
            System.out.println();
        }

    }
}
```

24. Write a program to print the following pattern:

1

22

333

4444

55555

```java
public class Star4 {
    public static void main(String[] args) {
        for (int i = 1; i <= 5; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(+i);
            }
            System.out.println();
        }
    }
}
```

25.Write a program to print the following pattern:

1

12

123

1234

12345

```java
public class Star2 {
    public static void main(String[] args) {
        for (int i = 1; i <= 5; i++) {
            for (int j = 1; j <= i; j++) {
                System.out.print(+j);
            }
            System.out.println();
        }
    }
}
```

26 Write a program to print the following pattern:

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

```
public class Star6 {
    public static void main(String[] args) {
        int i, j, k = 1;
        for (i = 1; i <= 5; ++i) {
            for (j = 1; j < i + 1; j++) {
                System.out.print(k++ + " ");
            }
            System.out.println();
        }

    }
}
```