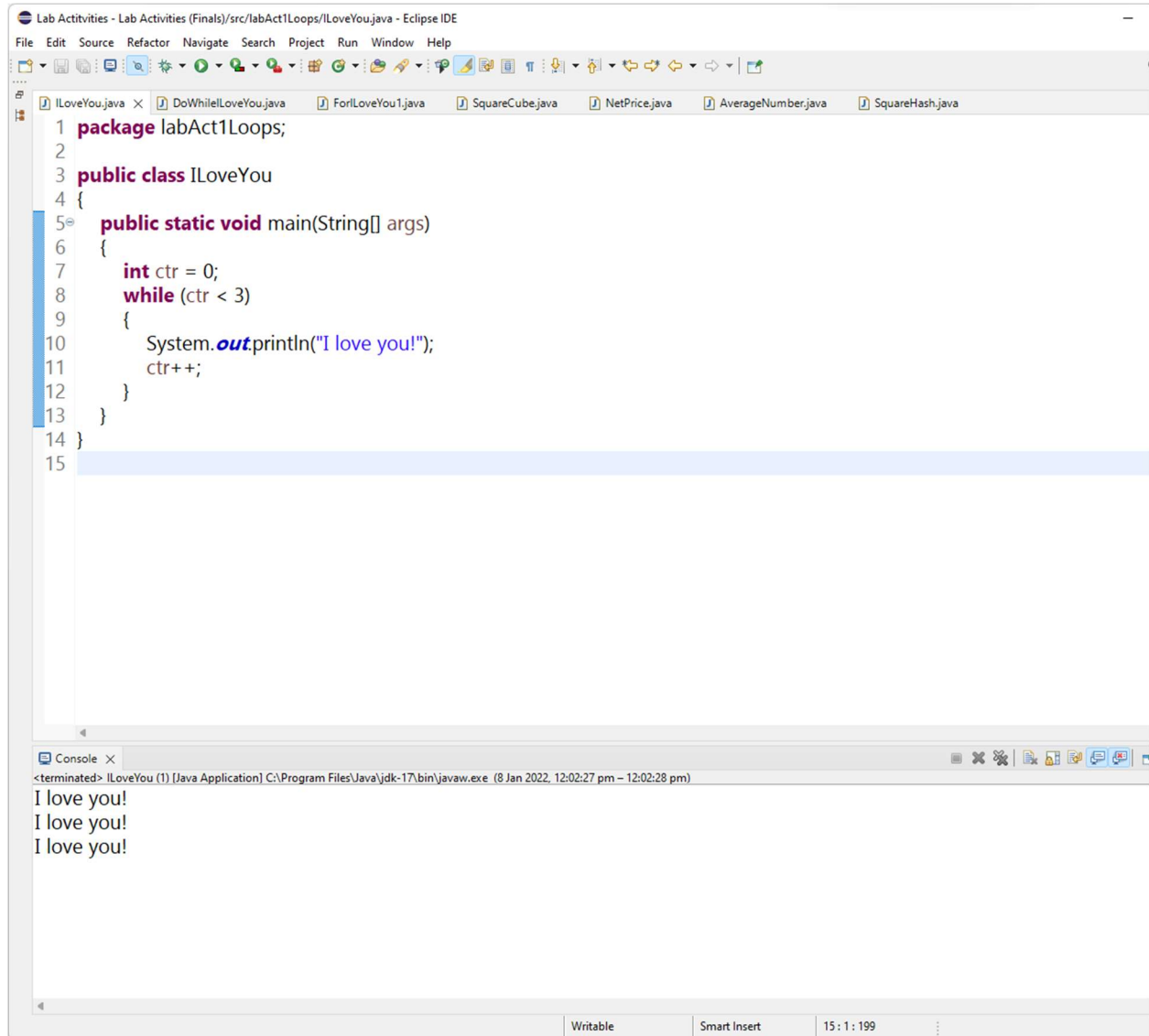


DOCUMENTATION REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

While Loop – IloveYou



The screenshot displays the Eclipse IDE interface. The main editor window shows a Java file named `ILoveYou.java` with the following code:

```
1 package labAct1Loops;
2
3 public class ILoveYou
4 {
5     public static void main(String[] args)
6     {
7         int ctr = 0;
8         while (ctr < 3)
9         {
10             System.out.println("I love you!");
11             ctr++;
12         }
13     }
14 }
15
```

The console window at the bottom shows the output of the program:

```
<terminated> ILoveYou (1) [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022, 12:02:27 pm - 12:02:28 pm)
I love you!
I love you!
I love you!
```

The status bar at the bottom indicates the file is writable, has smart insert enabled, and the cursor is at line 15, column 199.

Video Link:

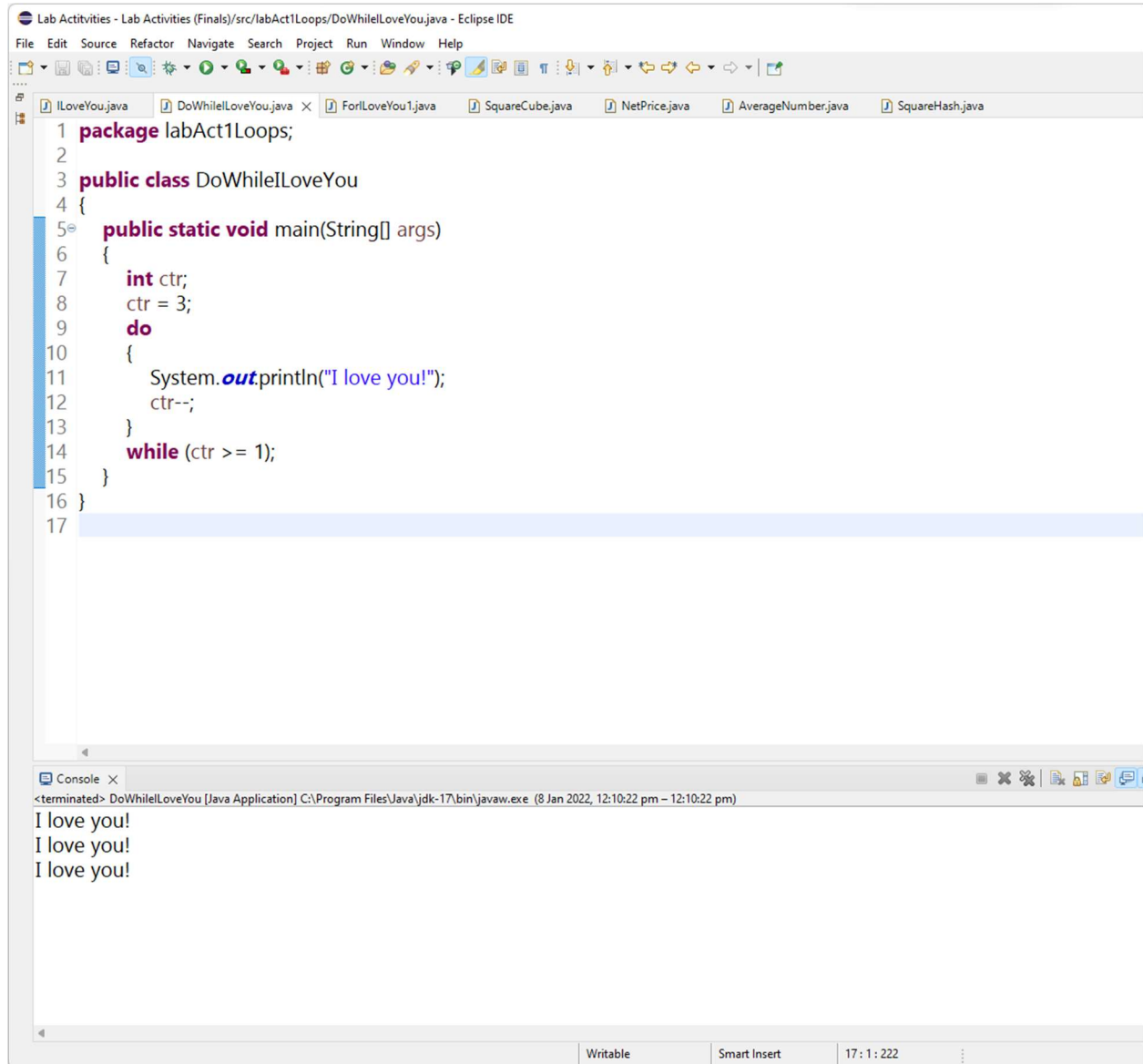
<https://drive.google.com/file/d/1c0WZYAGlZGLehMRxU8bkIN72Iz-BQxIQ/view?usp=sharing>

DOCUMENTATION REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

Do-while Loop – DoWhileILoveYou



The screenshot displays the Eclipse IDE interface. The main editor window shows the source code for `DoWhileILoveYou.java`. The code defines a package `labAct1Loops` and a public class `DoWhileILoveYou` with a `main` method. Inside the `main` method, an integer `ctr` is initialized to 3, and a `do-while` loop is used to print "I love you!" three times, decrementing `ctr` after each iteration. The console window at the bottom shows the output of the program, which is "I love you!" printed on three separate lines. The status bar at the bottom indicates the file is writable and the current time is 17:1:22.

```
1 package labAct1Loops;
2
3 public class DoWhileILoveYou
4 {
5     public static void main(String[] args)
6     {
7         int ctr;
8         ctr = 3;
9         do
10        {
11            System.out.println("I love you!");
12            ctr--;
13        }
14        while (ctr >= 1);
15    }
16 }
17
```

Console Output:

```
<terminated> DoWhileILoveYou [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022, 12:10:22 pm - 12:10:22 pm)
I love you!
I love you!
I love you!
```

Video Link:

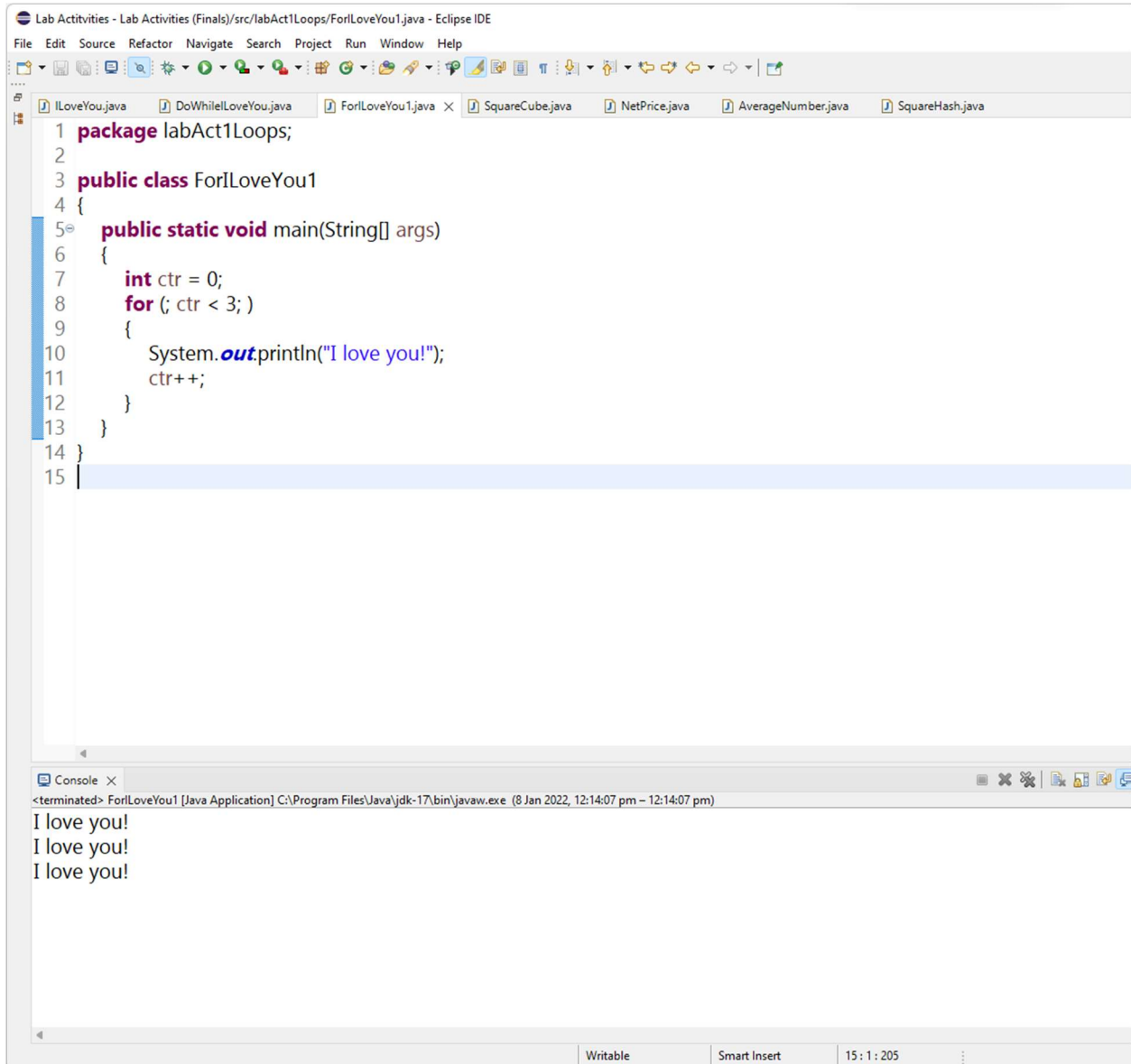
<https://drive.google.com/file/d/140xGMEXwDgbCU8yPqWrX745W6VpQ-uXo/view?usp=sharing>

DOCUMENTATION REPETITION AND CONTROL

Marasigan, Vem Aiensi A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

For Loop – ForILoveYou1



The screenshot displays the Eclipse IDE interface. The top menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations and development tools. The package explorer on the left shows a project named 'Lab Activities' with a sub-package 'labAct1Loops'. The editor window displays the source code for 'ForILoveYou1.java'. The code is as follows:

```
1 package labAct1Loops;
2
3 public class ForILoveYou1
4 {
5     public static void main(String[] args)
6     {
7         int ctr = 0;
8         for (; ctr < 3; )
9         {
10             System.out.println("I love you!");
11             ctr++;
12         }
13     }
14 }
15
```

The console window at the bottom shows the output of the program, which is three lines of 'I love you!'. The status bar at the bottom indicates the file is writable, smart insert is enabled, and the cursor is at line 15, column 205.

Video Link:

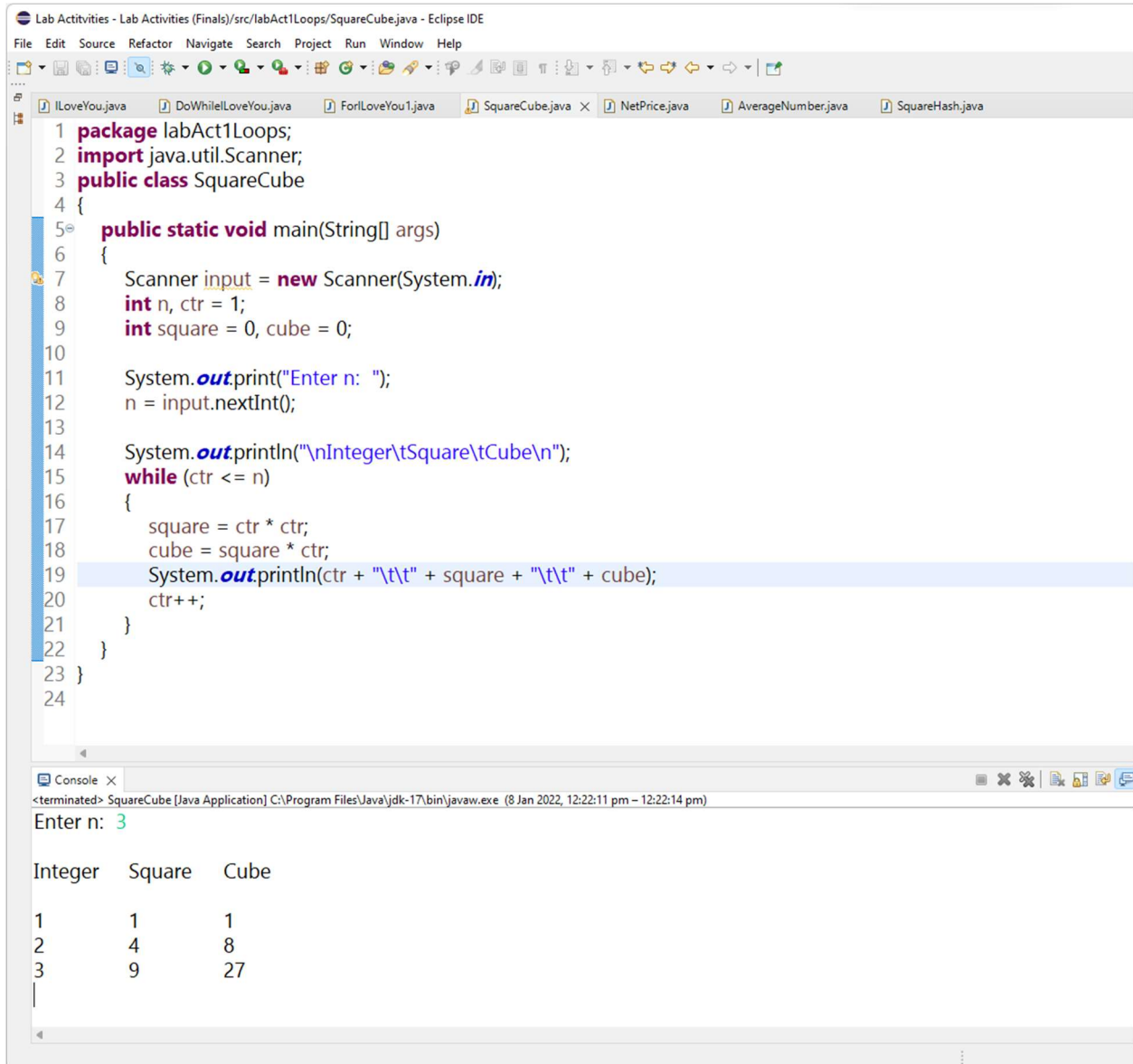
<https://drive.google.com/file/d/1-L5vrTf3sxJZowwQGpEnpA8dFoulbEhP/view?usp=sharing>

DOCUMENTATION
REPETITION AND CONTROL

Marasigan, Vem Aiensi A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

SquareCube



```
1 package labAct1Loops;
2 import java.util.Scanner;
3 public class SquareCube
4 {
5     public static void main(String[] args)
6     {
7         Scanner input = new Scanner(System.in);
8         int n, ctr = 1;
9         int square = 0, cube = 0;
10
11         System.out.print("Enter n: ");
12         n = input.nextInt();
13
14         System.out.println("\nInteger\tSquare\tCube\n");
15         while (ctr <= n)
16         {
17             square = ctr * ctr;
18             cube = square * ctr;
19             System.out.println(ctr + "\t\t" + square + "\t\t" + cube);
20             ctr++;
21         }
22     }
23 }
24
```

Console X

<terminated> SquareCube [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022, 12:22:11 pm – 12:22:14 pm)

Enter n: 3

Integer	Square	Cube
1	1	1
2	4	8
3	9	27

Video Link:

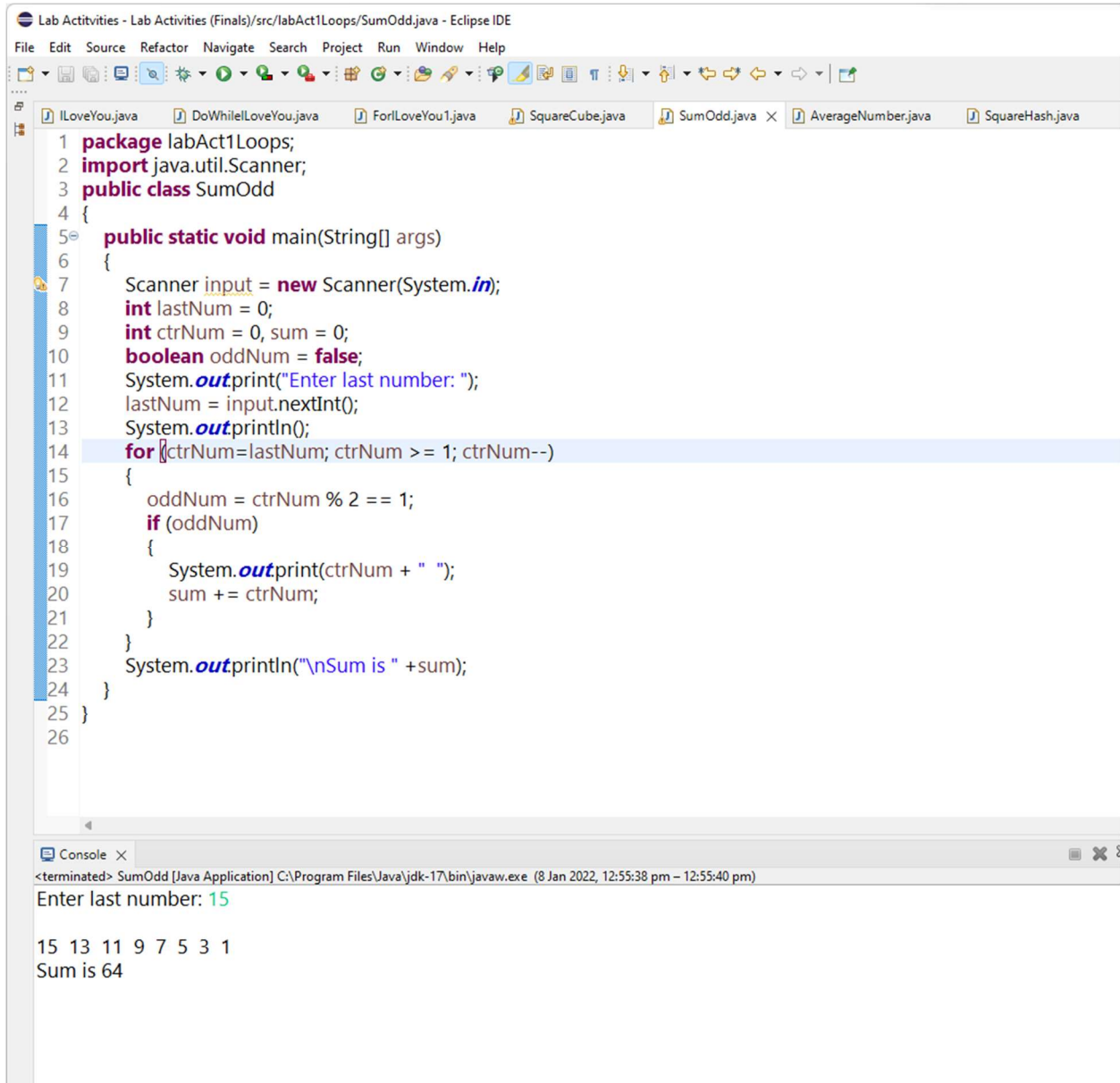
https://drive.google.com/file/d/1X-7YQrLGm82MQrCgkvXlXkSd_rMghDZc/view?usp=sharing

DOCUMENTATION
REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

SumOdd



The screenshot displays the Eclipse IDE interface. The top toolbar includes standard development tools like File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. Below the toolbar, several Java files are open in tabs: ILoveYou.java, DoWhileLoveYou.java, ForLoveYou1.java, SquareCube.java, SumOdd.java (active), AverageNumber.java, and SquareHash.java. The main editor window shows the code for SumOdd.java, which is a Java program to calculate the sum of odd numbers from 1 to a user-defined last number. The code uses a for loop to iterate from the last number down to 1, checking if each number is odd and adding it to a running sum. The console window at the bottom shows the program's execution: it prompts for the last number (15), displays the sequence of odd numbers (15 13 11 9 7 5 3 1), and finally outputs the sum (Sum is 64).

```
1 package labAct1Loops;
2 import java.util.Scanner;
3 public class SumOdd
4 {
5     public static void main(String[] args)
6     {
7         Scanner input = new Scanner(System.in);
8         int lastNum = 0;
9         int ctrNum = 0, sum = 0;
10        boolean oddNum = false;
11        System.out.print("Enter last number: ");
12        lastNum = input.nextInt();
13        System.out.println();
14        for (ctrNum=lastNum; ctrNum >= 1; ctrNum--)
15        {
16            oddNum = ctrNum % 2 == 1;
17            if (oddNum)
18            {
19                System.out.print(ctrNum + " ");
20                sum += ctrNum;
21            }
22        }
23        System.out.println("\nSum is " + sum);
24    }
25 }
26
```

Console X
<terminated> SumOdd [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022, 12:55:38 pm – 12:55:40 pm)
Enter last number: 15

15 13 11 9 7 5 3 1
Sum is 64

Video Link:

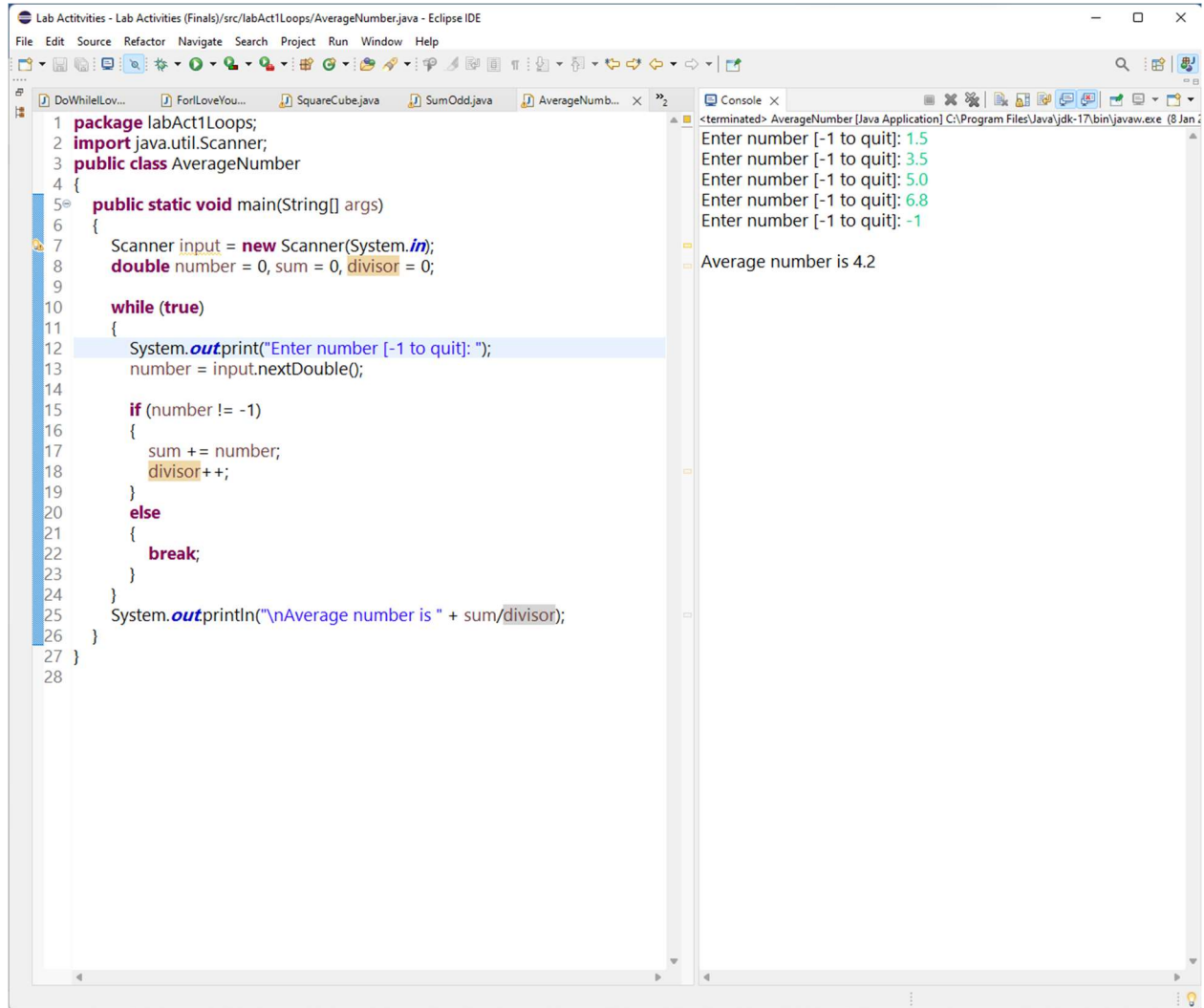
<https://drive.google.com/file/d/1YIU9G79ZvONLftLRH-ivg1hWNVSvlhTu/view?usp=sharing>

DOCUMENTATION REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

Break Statement - AverageNumber



The screenshot displays the Eclipse IDE interface. The left pane shows the source code for `AverageNumber.java`. The code uses a `while(true)` loop to repeatedly prompt the user for numbers. A `break;` statement is used to exit the loop when the user enters `-1`. The right pane shows the console output, which displays the prompts and the final average calculation.

```
1 package labAct1Loops;
2 import java.util.Scanner;
3 public class AverageNumber
4 {
5     public static void main(String[] args)
6     {
7         Scanner input = new Scanner(System.in);
8         double number = 0, sum = 0, divisor = 0;
9
10        while (true)
11        {
12            System.out.print("Enter number [-1 to quit]: ");
13            number = input.nextDouble();
14
15            if (number != -1)
16            {
17                sum += number;
18                divisor++;
19            }
20            else
21            {
22                break;
23            }
24        }
25        System.out.println("\nAverage number is " + sum/divisor);
26    }
27 }
28
```

Console Output:

```
<terminated> AverageNumber [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022)
Enter number [-1 to quit]: 1.5
Enter number [-1 to quit]: 3.5
Enter number [-1 to quit]: 5.0
Enter number [-1 to quit]: 6.8
Enter number [-1 to quit]: -1
Average number is 4.2
```

Video Link:

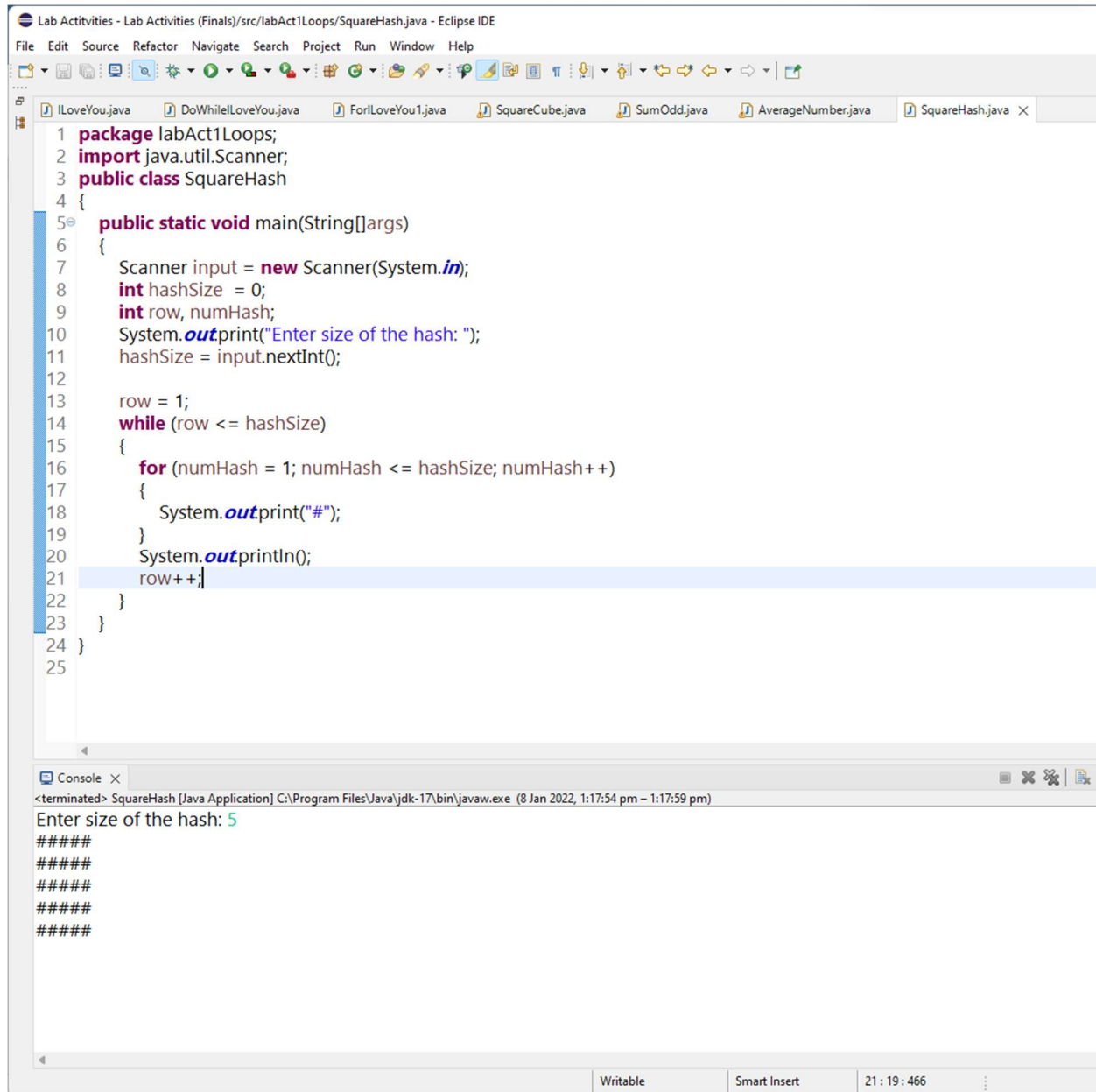
https://drive.google.com/file/d/187738_vEFuwAbo4TT0Vhot3LeMJiuYEo/view?usp=sharing

DOCUMENTATION
REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

Loop within a Loop – SquareHash



The screenshot displays the Eclipse IDE interface. The main editor window shows the source code for `SquareHash.java`. The code is as follows:

```
1 package labAct1Loops;
2 import java.util.Scanner;
3 public class SquareHash
4 {
5     public static void main(String[] args)
6     {
7         Scanner input = new Scanner(System.in);
8         int hashSize = 0;
9         int row, numHash;
10        System.out.print("Enter size of the hash: ");
11        hashSize = input.nextInt();
12
13        row = 1;
14        while (row <= hashSize)
15        {
16            for (numHash = 1; numHash <= hashSize; numHash++)
17            {
18                System.out.print("#");
19            }
20            System.out.println();
21            row++;
22        }
23    }
24 }
25
```

The console window at the bottom shows the execution output:

```
<terminated> SquareHash [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022, 1:17:54 pm - 1:17:59 pm)
Enter size of the hash: 5
#####
#####
#####
#####
#####
```

The status bar at the bottom indicates the file is writable, in smart insert mode, and the cursor is at line 21, column 19.

Video Link:

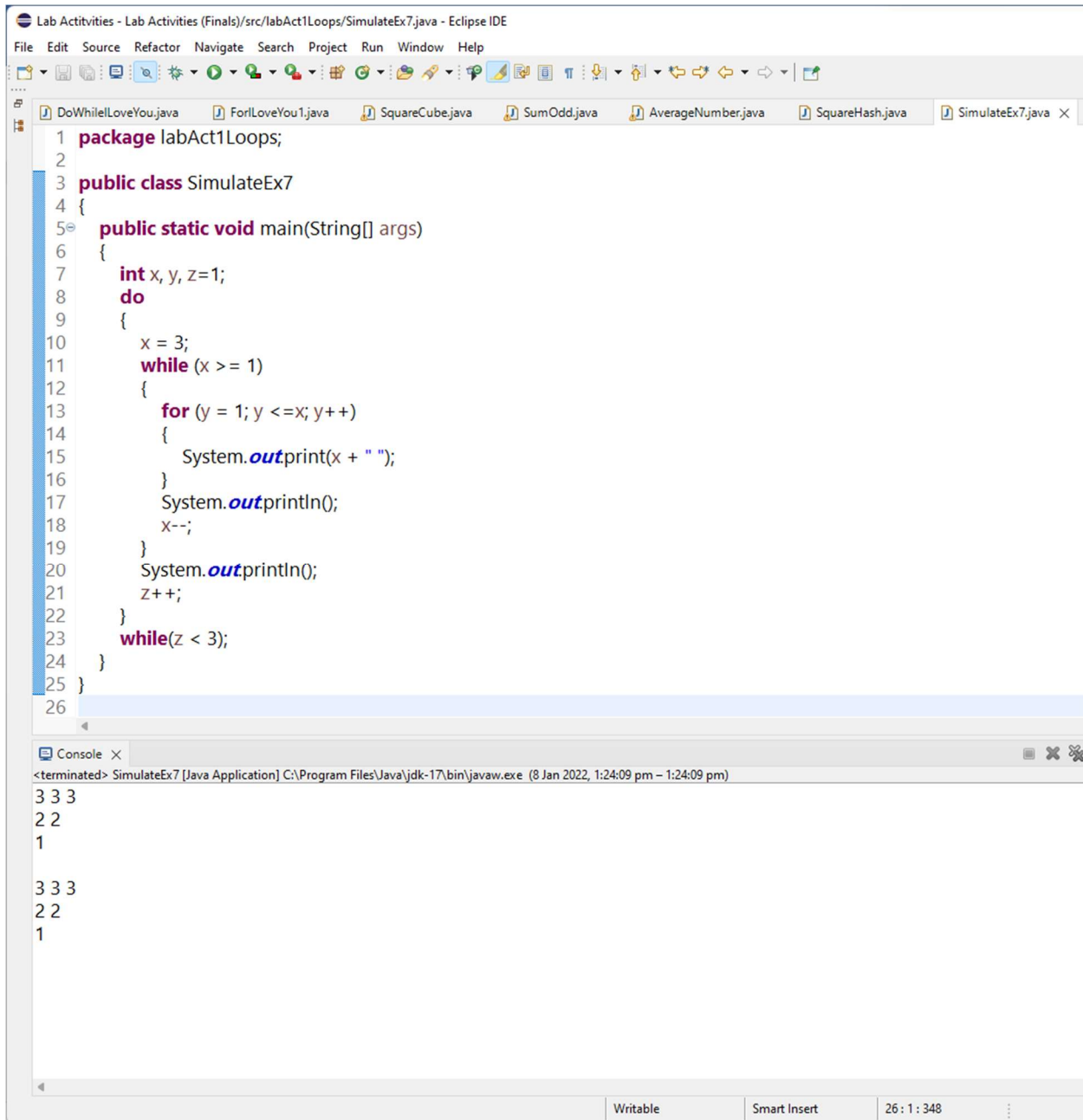
<https://drive.google.com/file/d/1jdWjuiVepyJeiDERk-kAupJ52mZT-a7K/view?usp=sharing>

DOCUMENTATION REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

SimulateEx7



The screenshot displays the Eclipse IDE interface. The top toolbar includes standard development tools like File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. Below the toolbar, a series of tabs represent open files: DoWhileLoveYou.java, ForLoveYou1.java, SquareCube.java, SumOdd.java, AverageNumber.java, SquareHash.java, and SimulateEx7.java. The SimulateEx7.java tab is active, showing the following Java code:

```
1 package labAct1Loops;
2
3 public class SimulateEx7
4 {
5     public static void main(String[] args)
6     {
7         int x, y, z=1;
8         do
9         {
10             x = 3;
11             while (x >= 1)
12             {
13                 for (y = 1; y <=x; y++)
14                 {
15                     System.out.print(x + " ");
16                 }
17                 System.out.println();
18                 x--;
19             }
20             System.out.println();
21             z++;
22         }
23         while(z < 3);
24     }
25 }
26
```

Below the code editor, the Console window is open, showing the output of the program. It displays two identical blocks of output, each consisting of three lines: "3 3 3", "2 2", and "1". The first block is followed by a blank line. The console title bar indicates the application is "SimulateEx7 [Java Application]" and the command used is "C:\Program Files\Java\jdk-17\bin\javaw.exe (8 Jan 2022, 1:24:09 pm - 1:24:09 pm)".

Video Link:

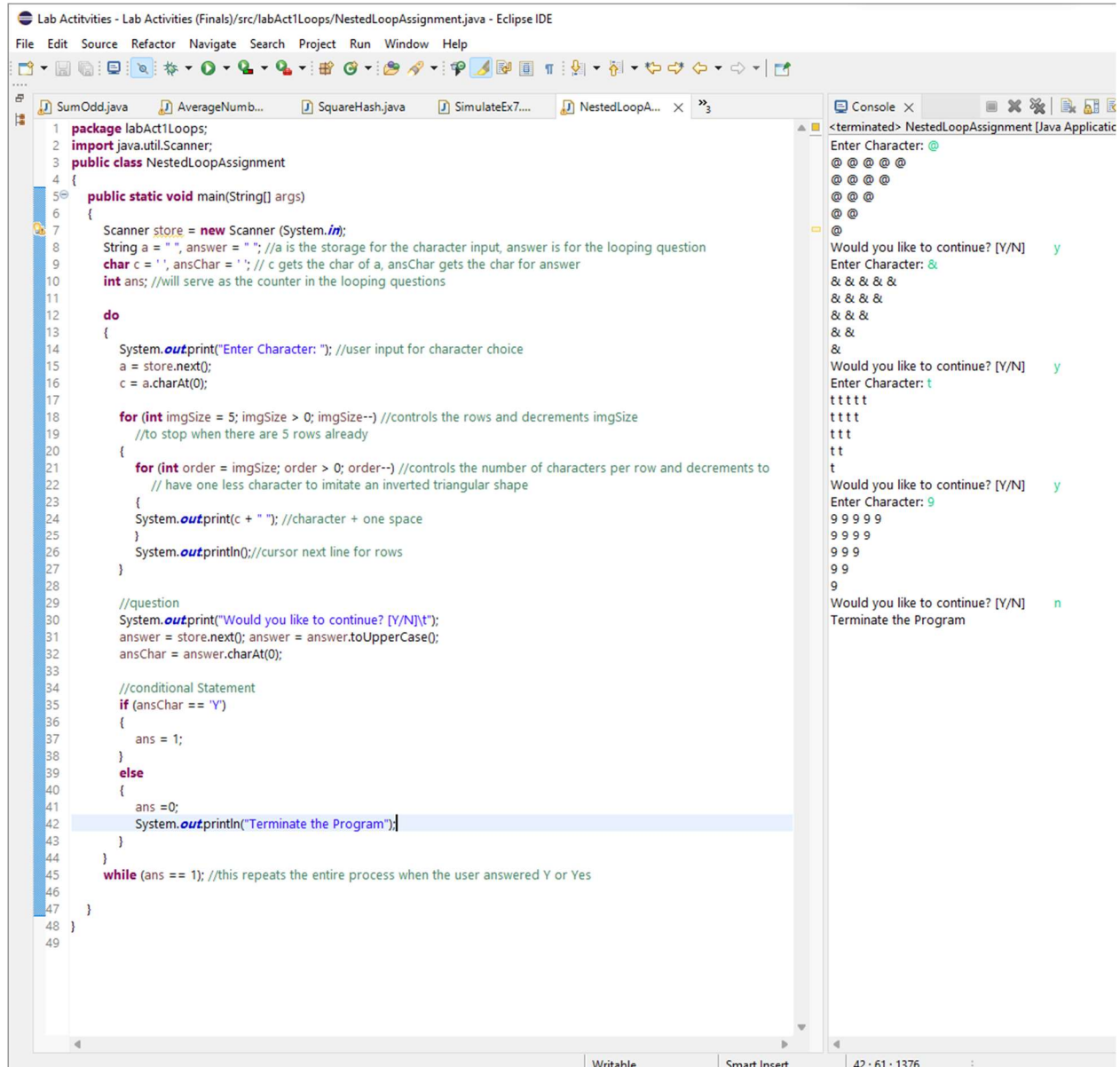
<https://drive.google.com/file/d/1W6btOX77c2Kq5KbeS7r3EsEGtIQKmNfg/view?usp=sharing>

DOCUMENTATION REPETITION AND CONTROL

Marasigan, Vem Aiens A.
1-BSEMC-.1
BS Computer Science

Nov 28, 2021 (Jan 8, 2022)

Nested Loop – Inverted Triangle



```
1 package labAct1Loops;
2 import java.util.Scanner;
3 public class NestedLoopAssignment
4 {
5     public static void main(String[] args)
6     {
7         Scanner store = new Scanner (System.in);
8         String a = " ", answer = " "; //a is the storage for the character input, answer is for the looping question
9         char c = ' ', ansChar = ' '; // c gets the char of a, ansChar gets the char for answer
10        int ans; //will serve as the counter in the looping questions
11
12        do
13        {
14            System.out.print("Enter Character: "); //user input for character choice
15            a = store.next();
16            c = a.charAt(0);
17
18            for (int imgSize = 5; imgSize > 0; imgSize--) //controls the rows and decrements imgSize
19                //to stop when there are 5 rows already
20            {
21                for (int order = imgSize; order > 0; order--) //controls the number of characters per row and decrements to
22                    // have one less character to imitate an inverted triangular shape
23                {
24                    System.out.print(c + " "); //character + one space
25                }
26                System.out.println(); //cursor next line for rows
27            }
28
29            //question
30            System.out.print("Would you like to continue? [Y/N]\n");
31            answer = store.next(); answer = answer.toUpperCase();
32            ansChar = answer.charAt(0);
33
34            //conditional Statement
35            if (ansChar == 'Y')
36            {
37                ans = 1;
38            }
39            else
40            {
41                ans = 0;
42                System.out.println("Terminate the Program");
43            }
44        }
45        while (ans == 1); //this repeats the entire process when the user answered Y or Yes
46    }
47 }
48
49
```

Console Output:

```
<terminated> NestedLoopAssignment [Java Applicati
Enter Character: @
@@ @@@@
@@ @@@
@@ @@
@@ @
@@
@
Would you like to continue? [Y/N] y
Enter Character: &
&&&&&&
&&&&
&&&
&&
&
Would you like to continue? [Y/N] y
Enter Character: t
ttttt
tttt
ttt
tt
t
Would you like to continue? [Y/N] y
Enter Character: 9
99999
9999
999
99
9
Would you like to continue? [Y/N] n
Terminate the Program
```

Video Link:

https://drive.google.com/file/d/19M5_lIlgxsofLqk7s55-eKlk3yoYVFmN/view?usp=sharing