|  |  |  |
| --- | --- | --- |
| **Lesson Title: Sequential and Conditional Control Structures Lesson Objectives:**  At the end of the lesson, you shoud be able to:   1. understand the step – by – step process during program execution; 2. write Java programs involving sequential and conditional control structures. |  | **Materials:**  Student Activity Sheets **References:**  Daniel, Liang. Introduction to Java Programming 7th Edition.  Java Tutorial  @www.w3schools.com |

|  |
| --- |
| Productivity Tip:  ***“Turn – off notifications and stay focus.”*** |

# A. LESSON PREVIEW/REVIEW

1. Introduction (2 mins)

Today is another brand new day. Let’s make it an awesome day by learning sequential and conditional control structures using the Java programming language. It is wise to learn these things because it gives us the flexibility to manipulate and specify what should be done when user tries to change or modify the data. Moreover, it gives us the power to specify when a situation occurs; particular action(s) will be implemented and executed. Therefore, at the end of this lesson, you should be able to:

1. understand the step – by – step process during program execution; 2. write Java programs involving sequential and conditional control structures.

1. Activity 1: What I Know Chart, Part 1 (3 mins)

I am writing three questions on the second column related to the topic for the day. I would like you to write in the first column whay you know about the question. Just left the third column blank at this time.

|  |  |  |
| --- | --- | --- |
| What I Know | Questions: | What I Learned (Activity 4) |
| Sequential, Selection, & Repetition | 1. What are the three types of control structures? |  |
| do-while loop | 2. What statement will be used to specify that a block of code will be executed if the condition is true? |  |
| nextInt() | 3. What method from the Utility package will be used to read user – input? |  |

# B.MAIN LESSON

1)

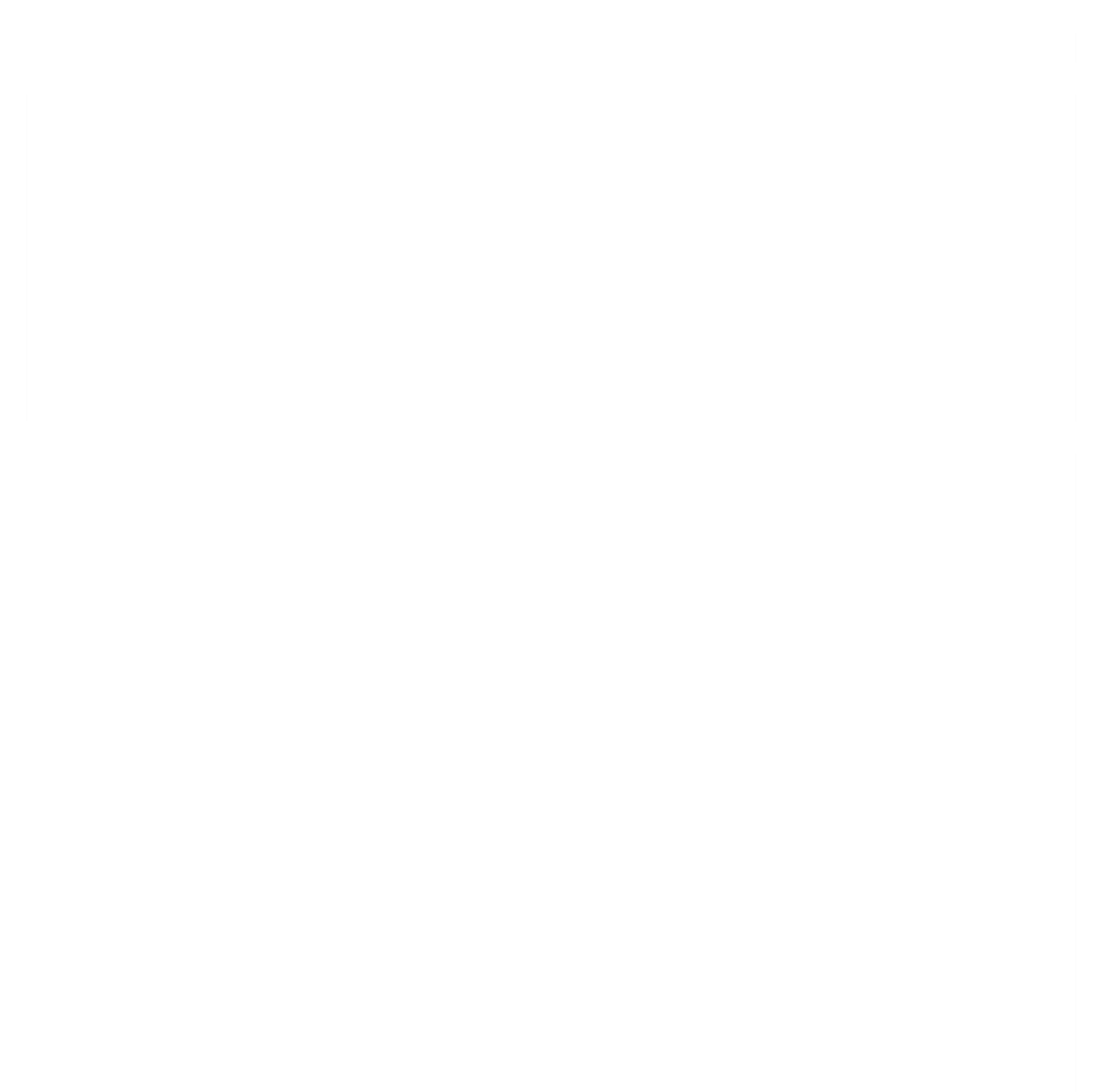
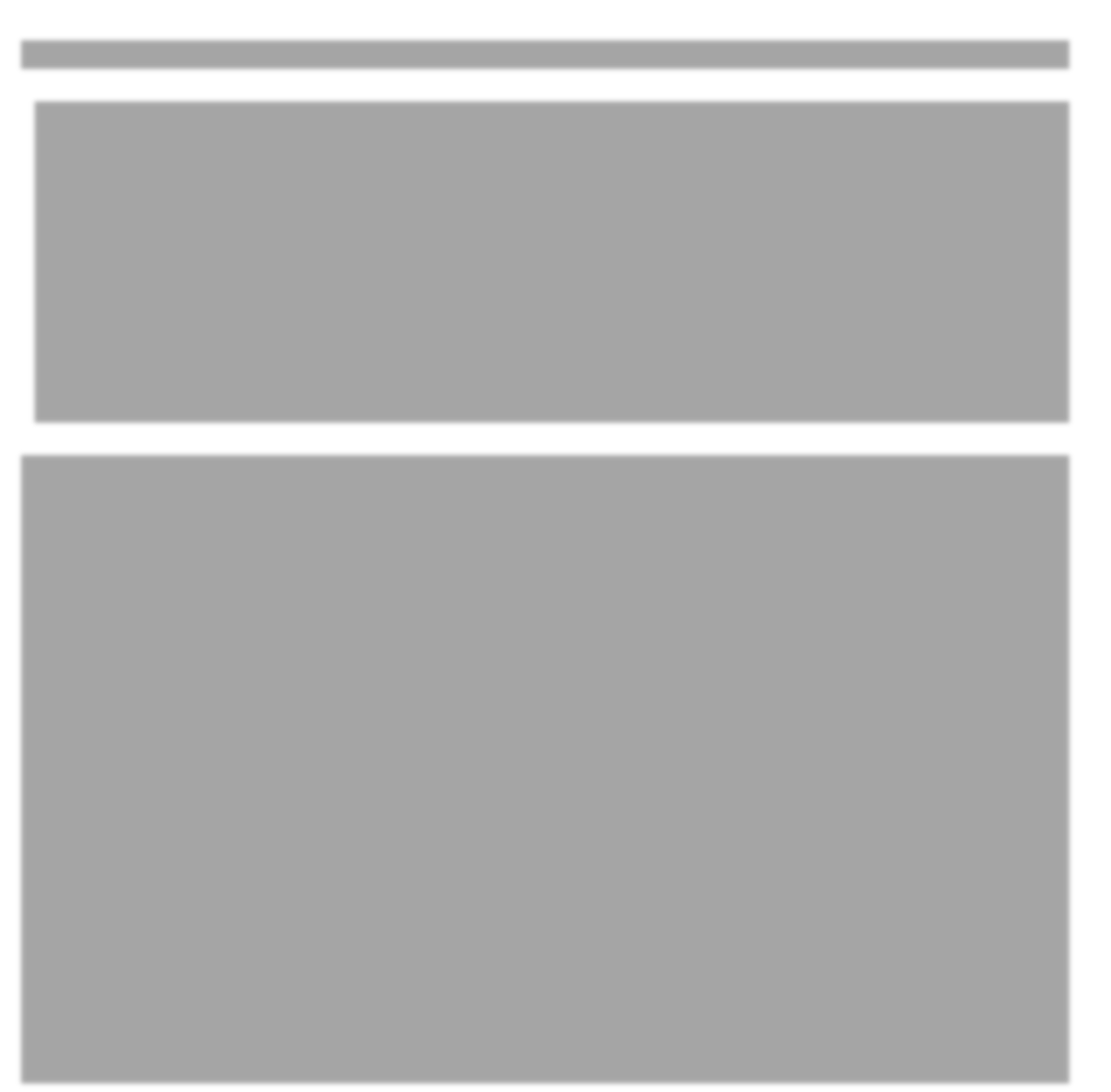
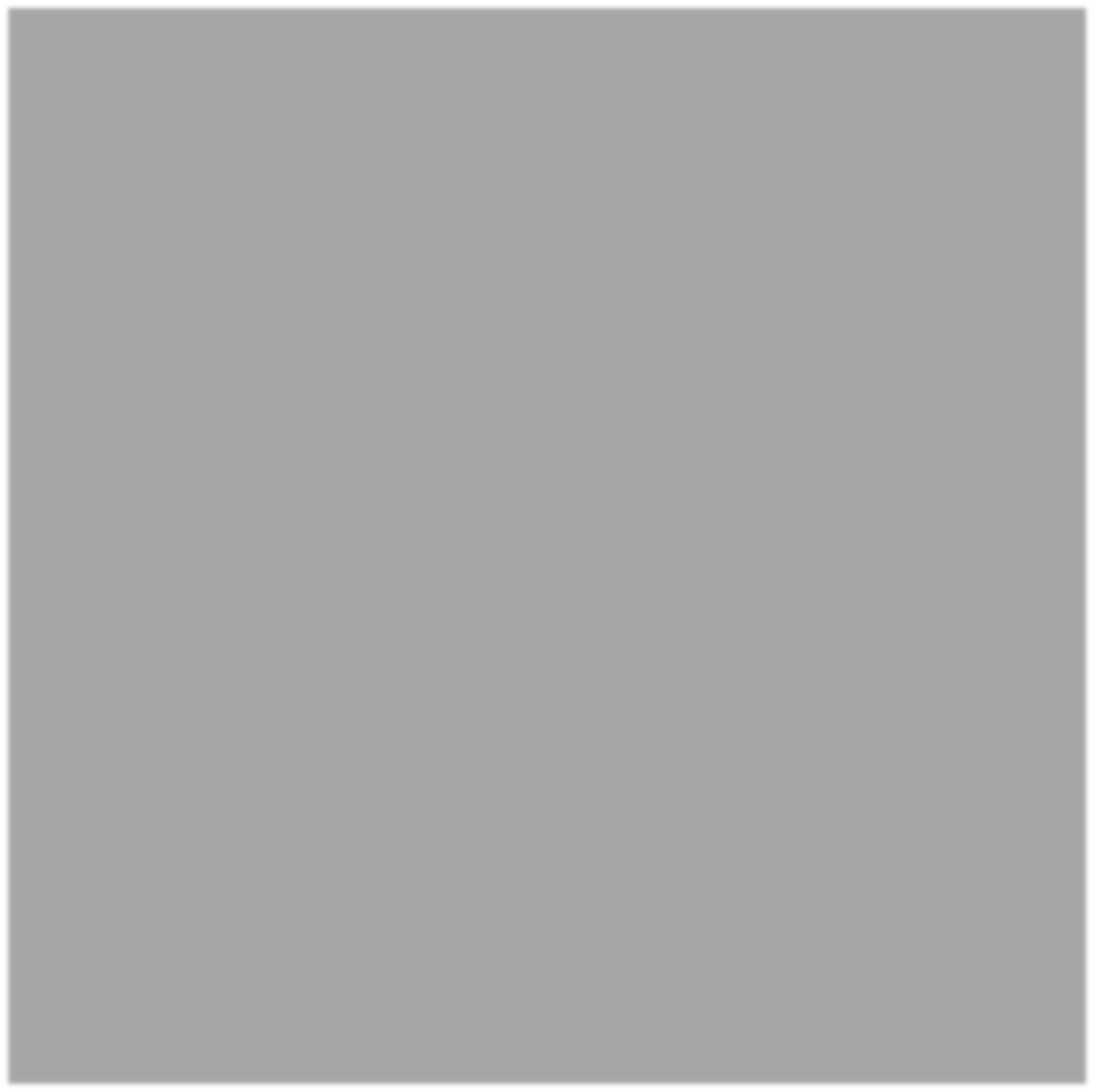
Act

ivity 2: Content Notes

)

mins

(45



**Three Basic Types of Control Structures**

1.

Sequential

:

default mode. Sequential execution of code statements (one line after

another)

--

like following a recipe.

2.

Selection

used for decisions, branching

:

--

choosing between 2 or more alternative paths.

In C++, these are the types of selection statements:

a. if

b. if/else

c. switch

3.

Repetition

:

used for looping, i.e. repeating a piece of code multiple times in a

row. In

JAVA, there are three types of loops:

while, do / while and for loop.

In today’s lesson, we will be focusing on sequential and selection / conditional control

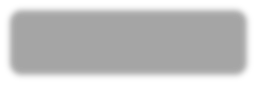
structures. Let us now recall the flowcharting lesson in Programming I.

**Sequential Con**

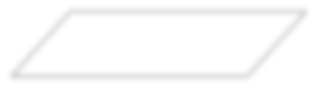
**trol Structure**

Example 1: Create a flowchart that would allow the user to input two integer values and

let’s compute and display the sum of these two values. Our solution will look like this one:

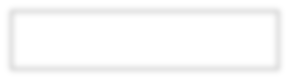


Start

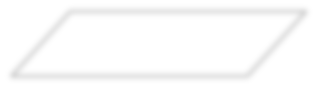


Input

num1, num2

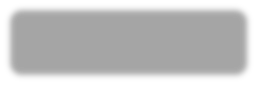


sum = num1 + num2

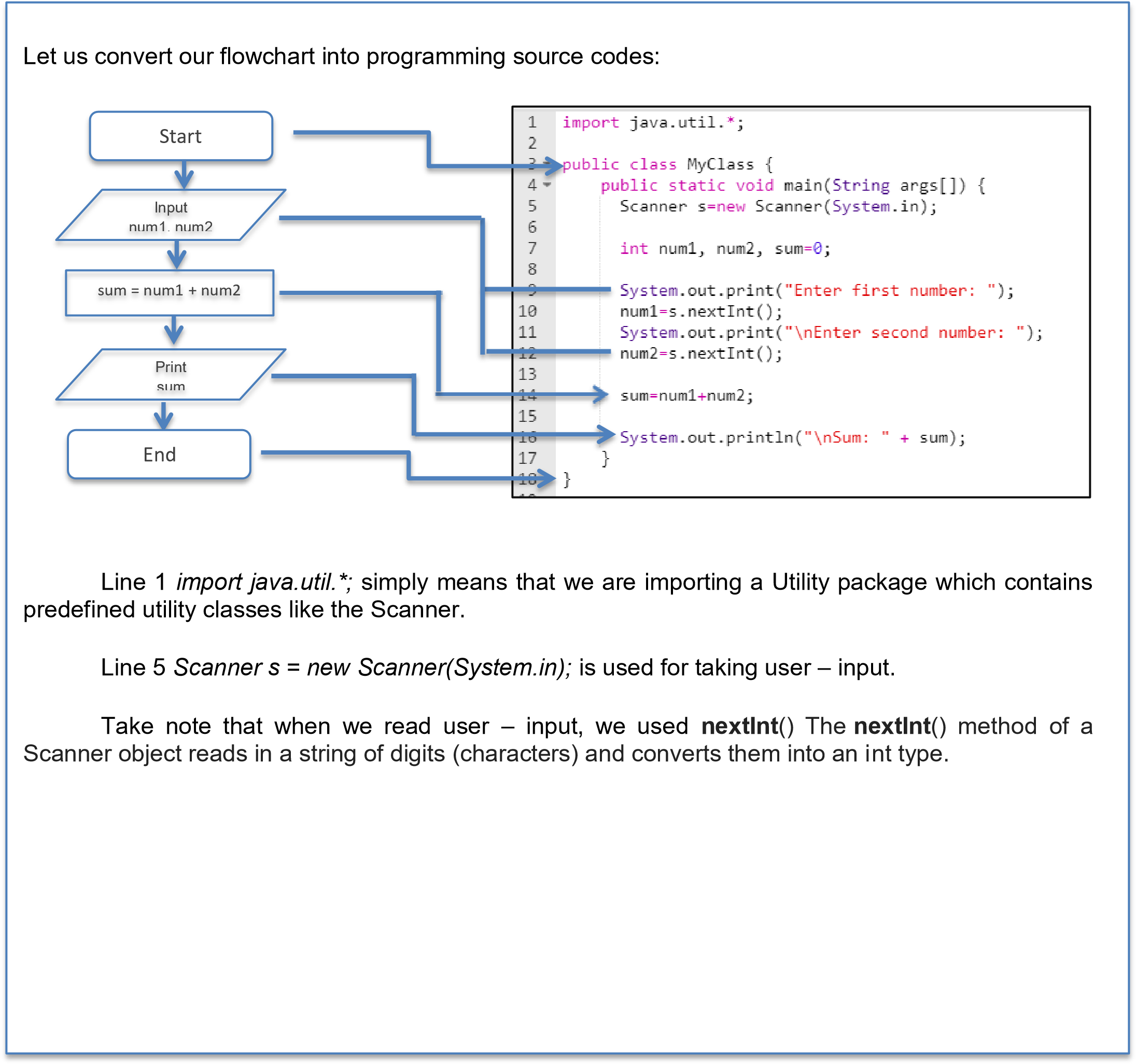


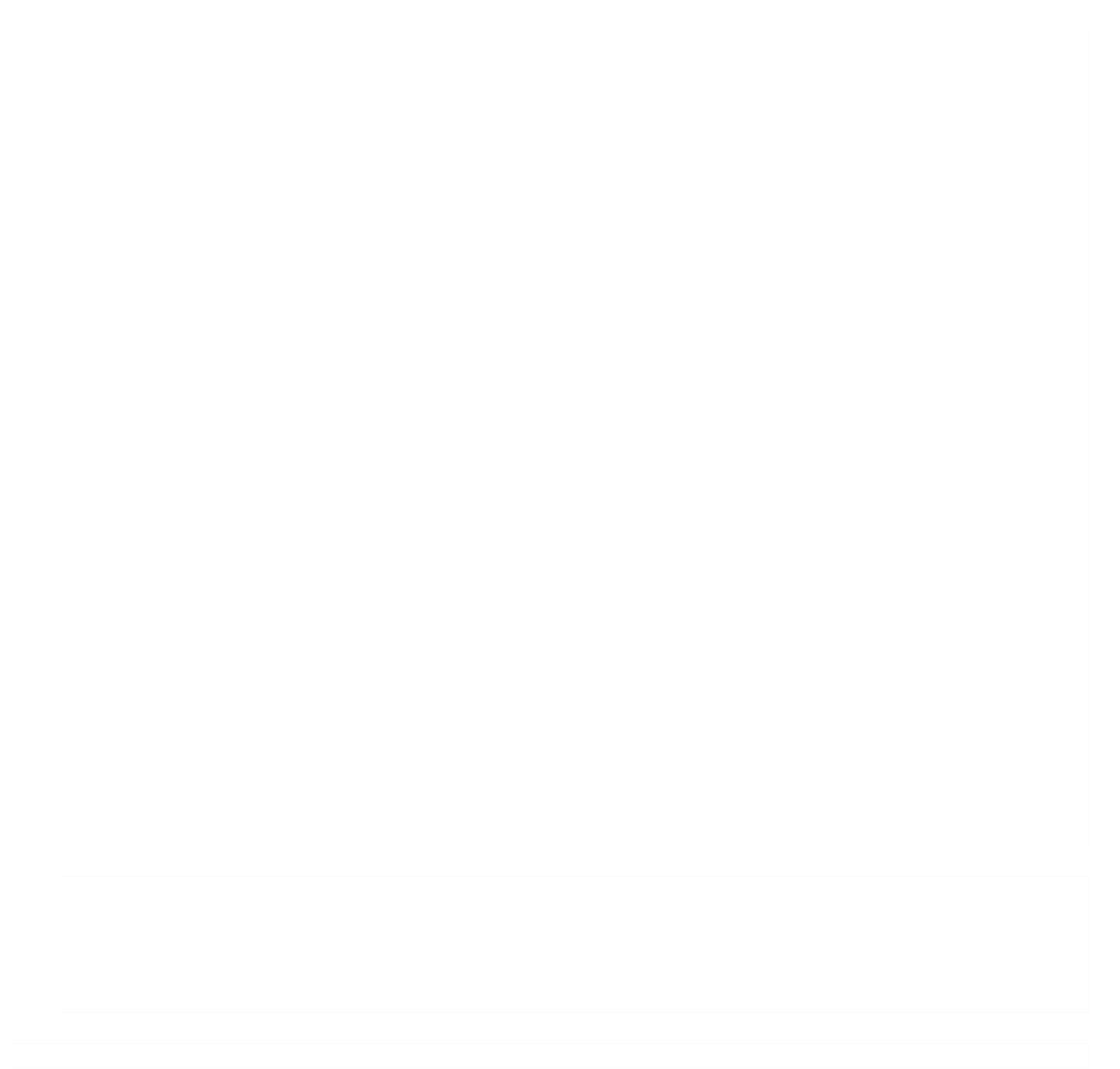
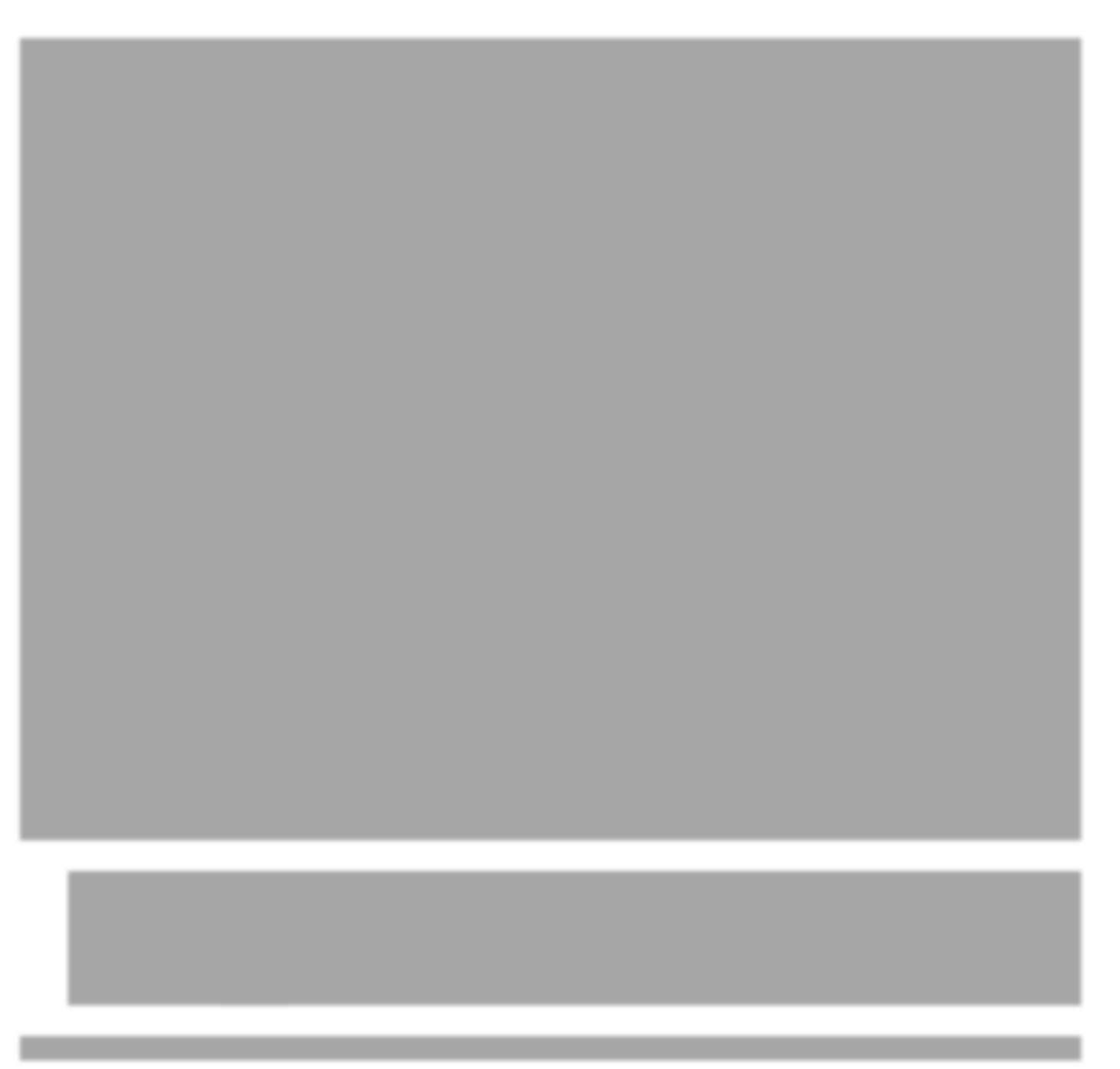
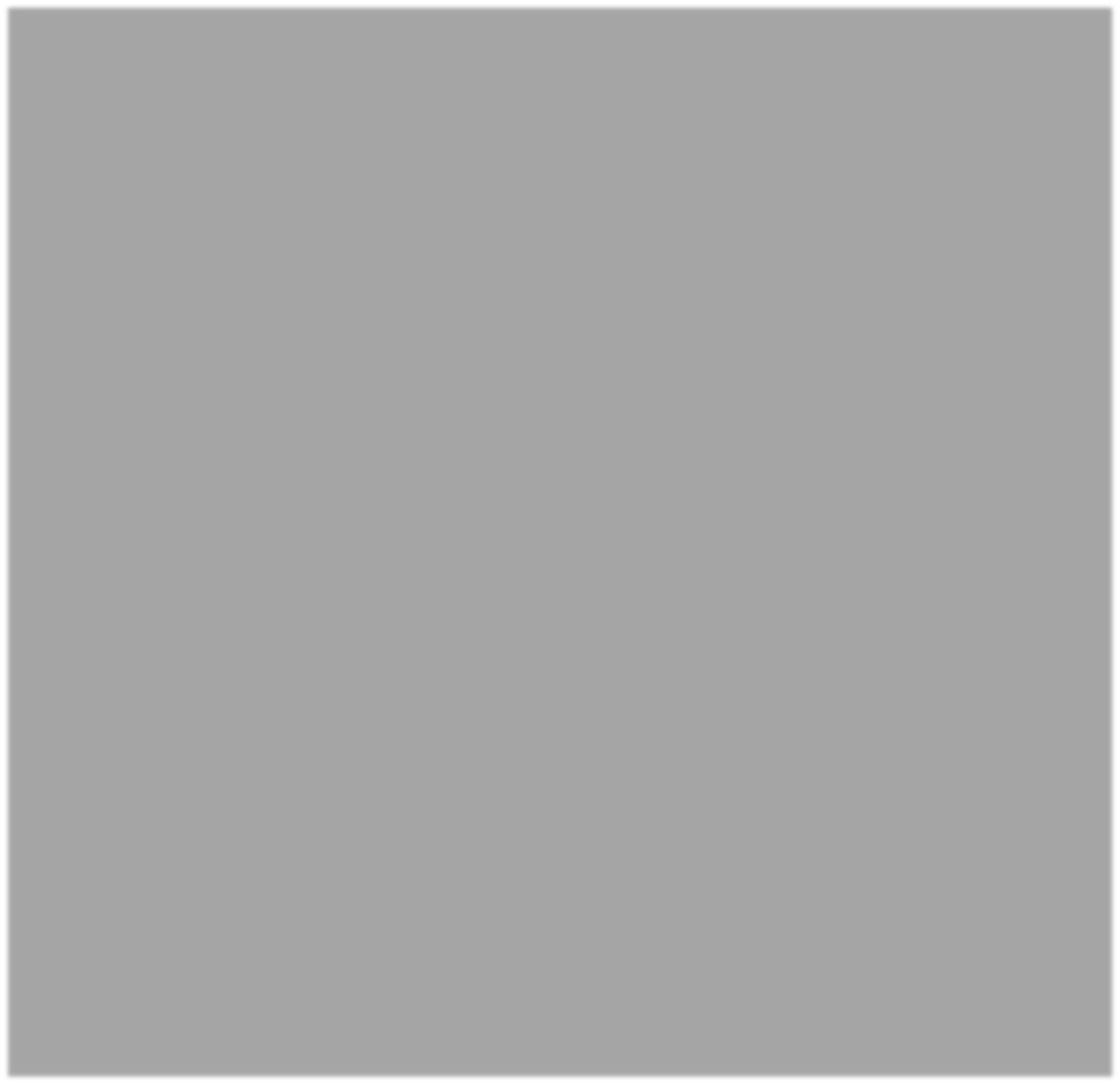
Print

sum



End





Example 2:

Create a Java program that would allow the user to input for the base and height of a

triangle. Compute and display the area.

Solution:

When we run our program, below is the sample output:

**Conditional Control Structures**

J

ava has the

following conditional statements:

1

.

Use

if

to specify a block of code to be executed, if a specified condition is true

2

.

Use

else

to specify a block of code to be executed, if the same condition is false

3

.

Use

else if

to specify a new condition to test,

if the first condition is false

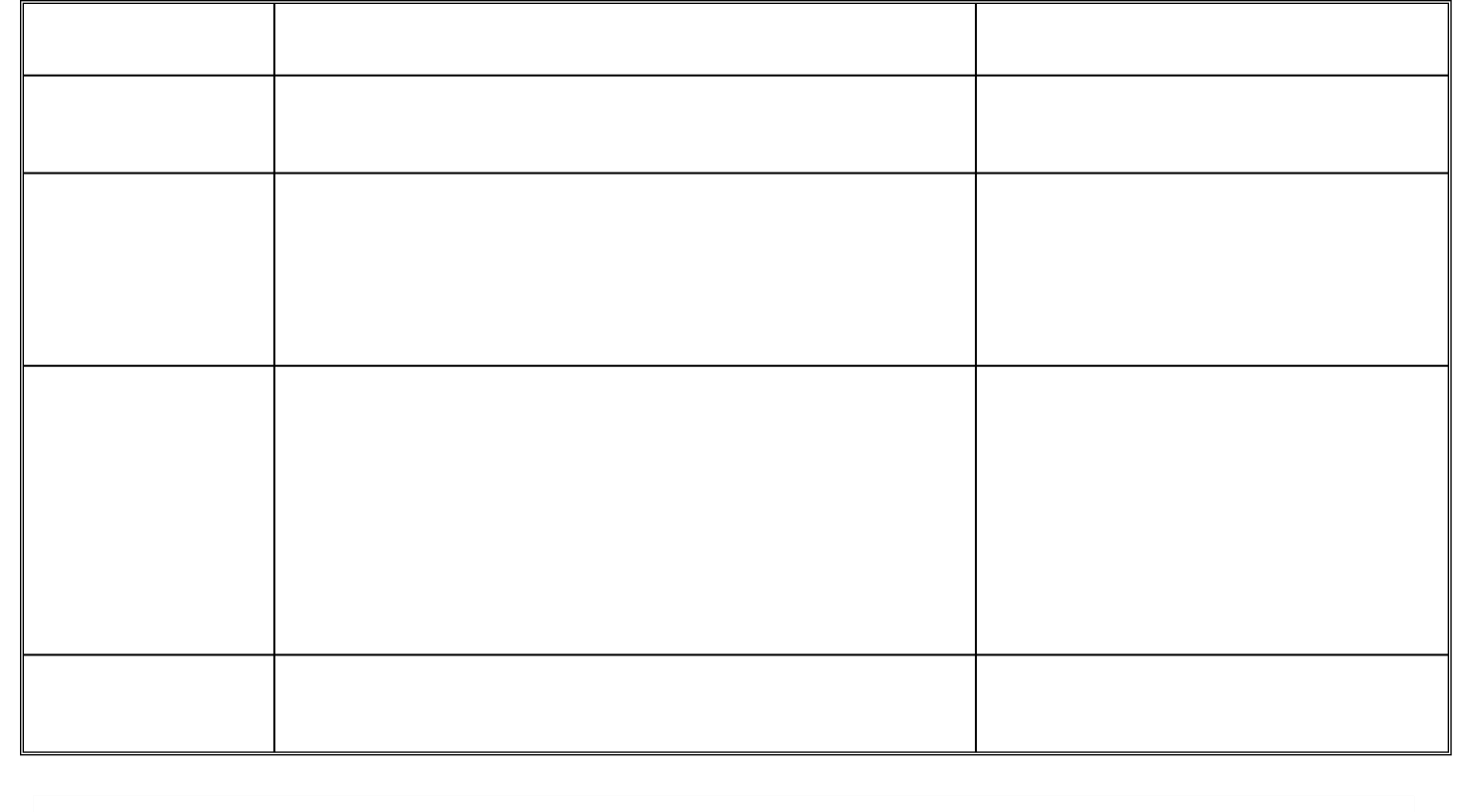
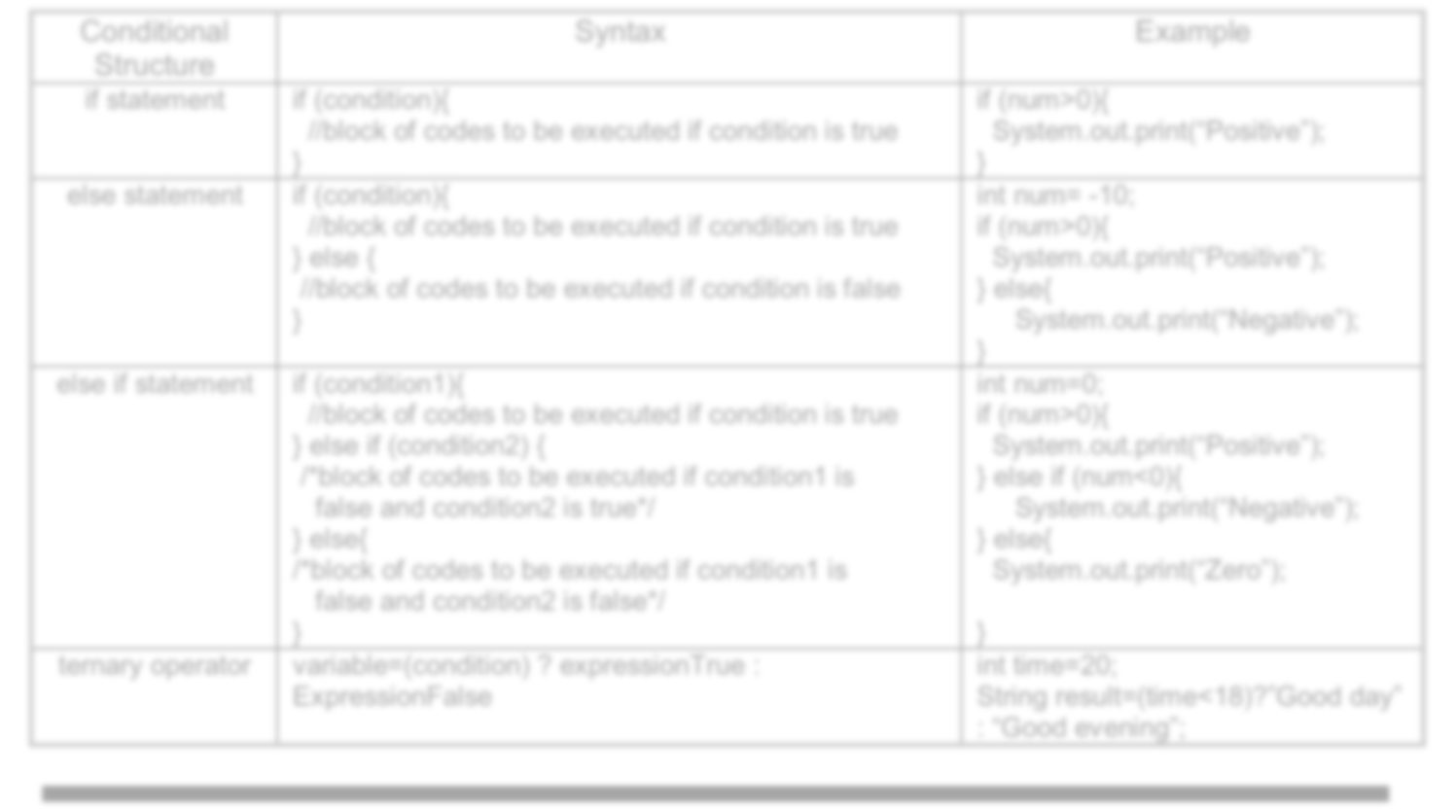
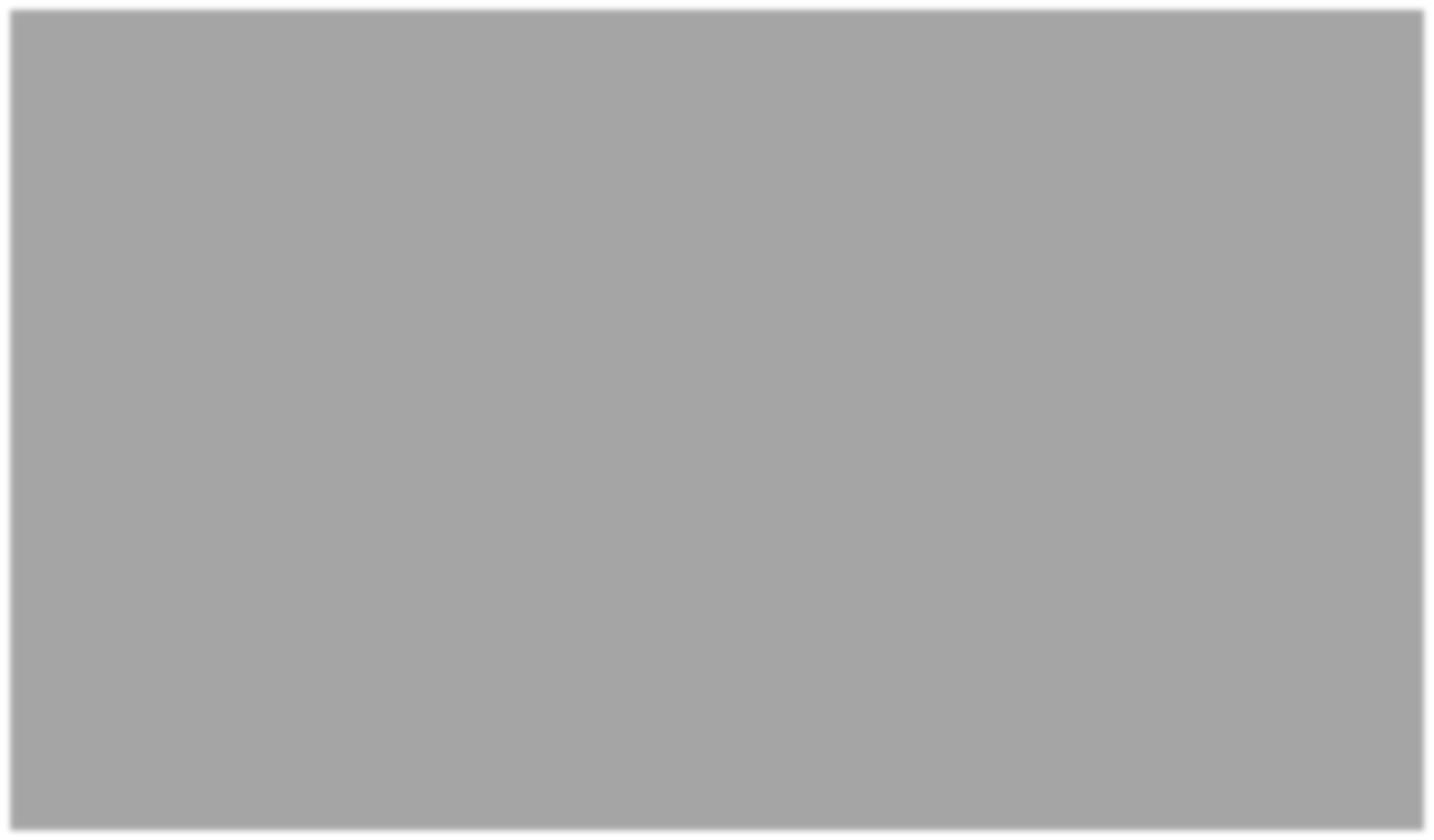
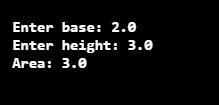
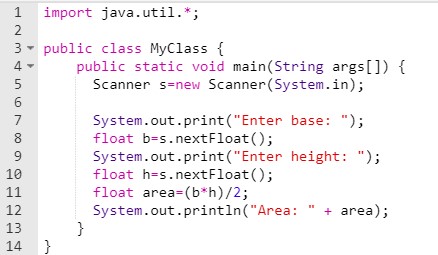
4

.

Use

switch

to specify many alternative blocks of code to be executed



Conditional

Structure

Syntax

Example

if statement

if (condition){

//block of codes to be executed if condition is true

}

if (num>0){

System.out.print(“Positive”);

}

else statement

if (condition){

//block of codes to be executed if condition is true

}

else

{

//block of codes to be executed if condition is false

}

int num=

-

10

;

if (num>0){

System.out.print(“Positive”);

}

else

{

System.out.print(“Negative”);

}

else if statement

if (condition1){

//block of codes to be executed if condition is true

}

else if (condition2)

{

/\*block of codes to be executed if condition1 is

false and condition2 is true\*/

}

else

{

/\*block of codes to be executed if condition1 is

false and condition2 is false\*/

}

int num=0;

if (num>0){

System.out.print(“Positive”);

}

else if (num

<0)

{

System.out.print(“Negative”);

}

else

{

System.out.print(“Zero”);

}

ternary operator

variable=

(

condition) ? expressionTrue :

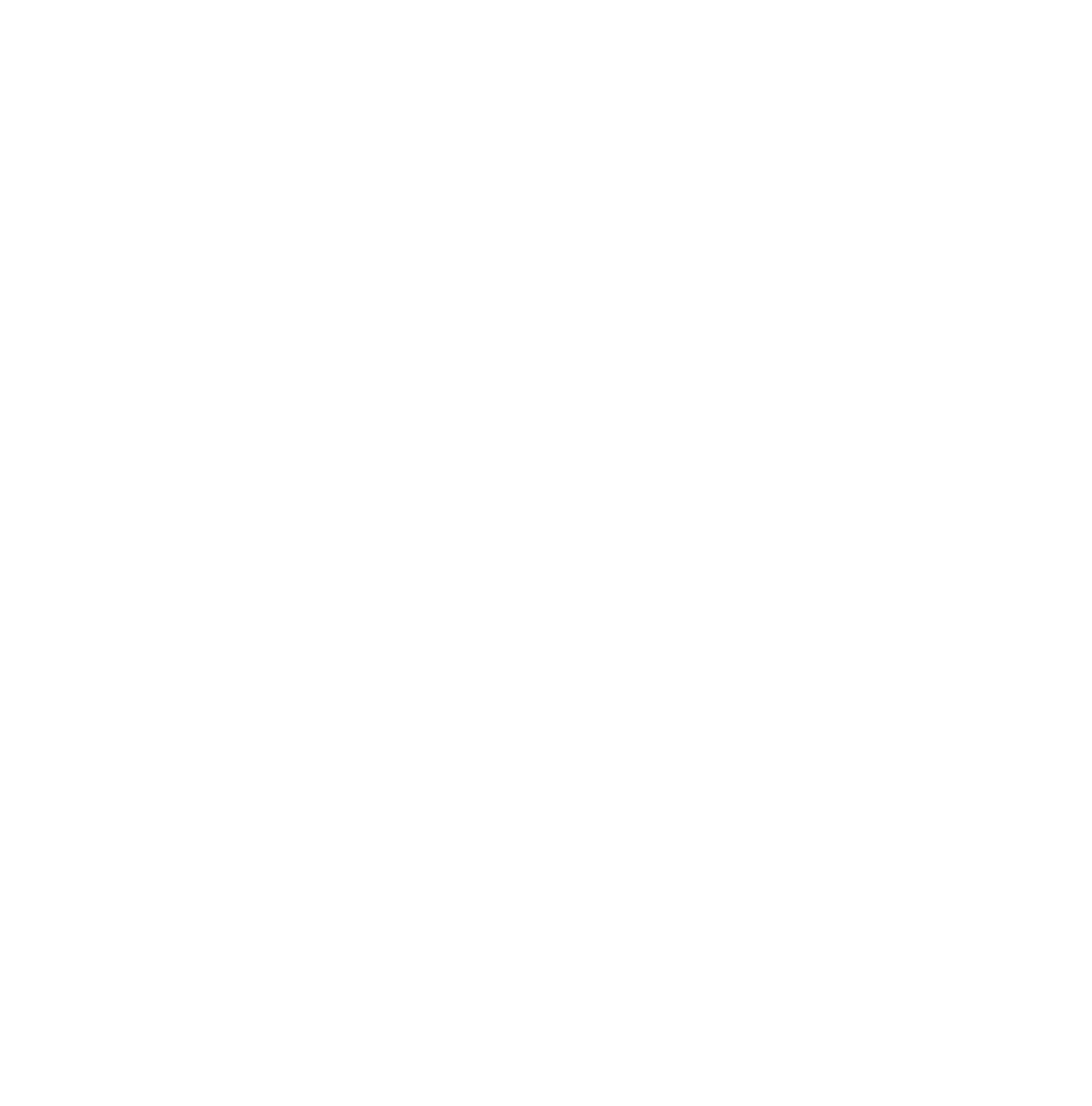
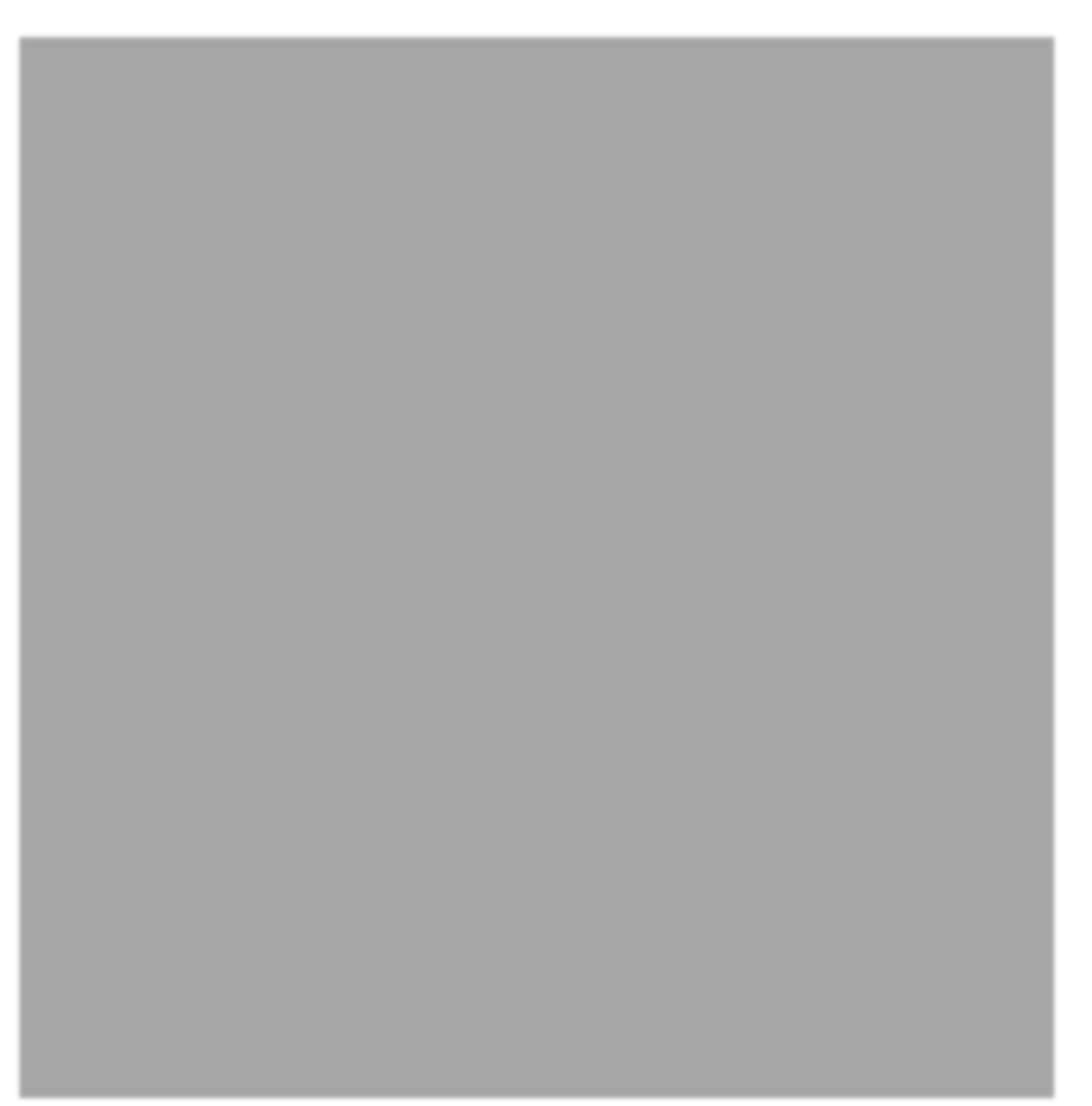
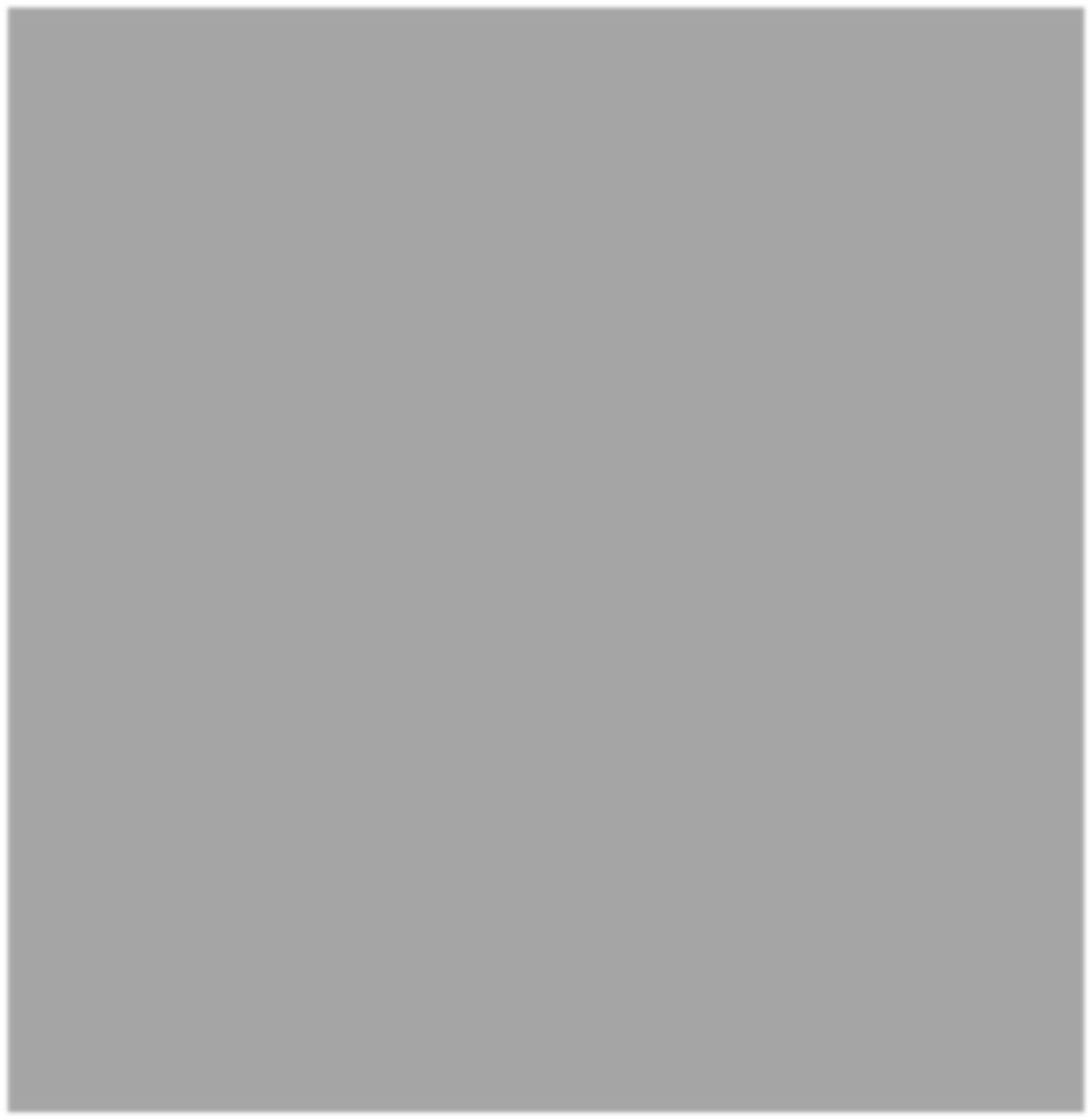
ExpressionFalse

int time=20;

String result=(time<18)?”Good day”

:

“Good evening”;



Example 1:

This program will ask the user to input the age and checks whether the entered age is greater

than or equals to 18. If this condition is met, then display the message “You are eligible for voting”

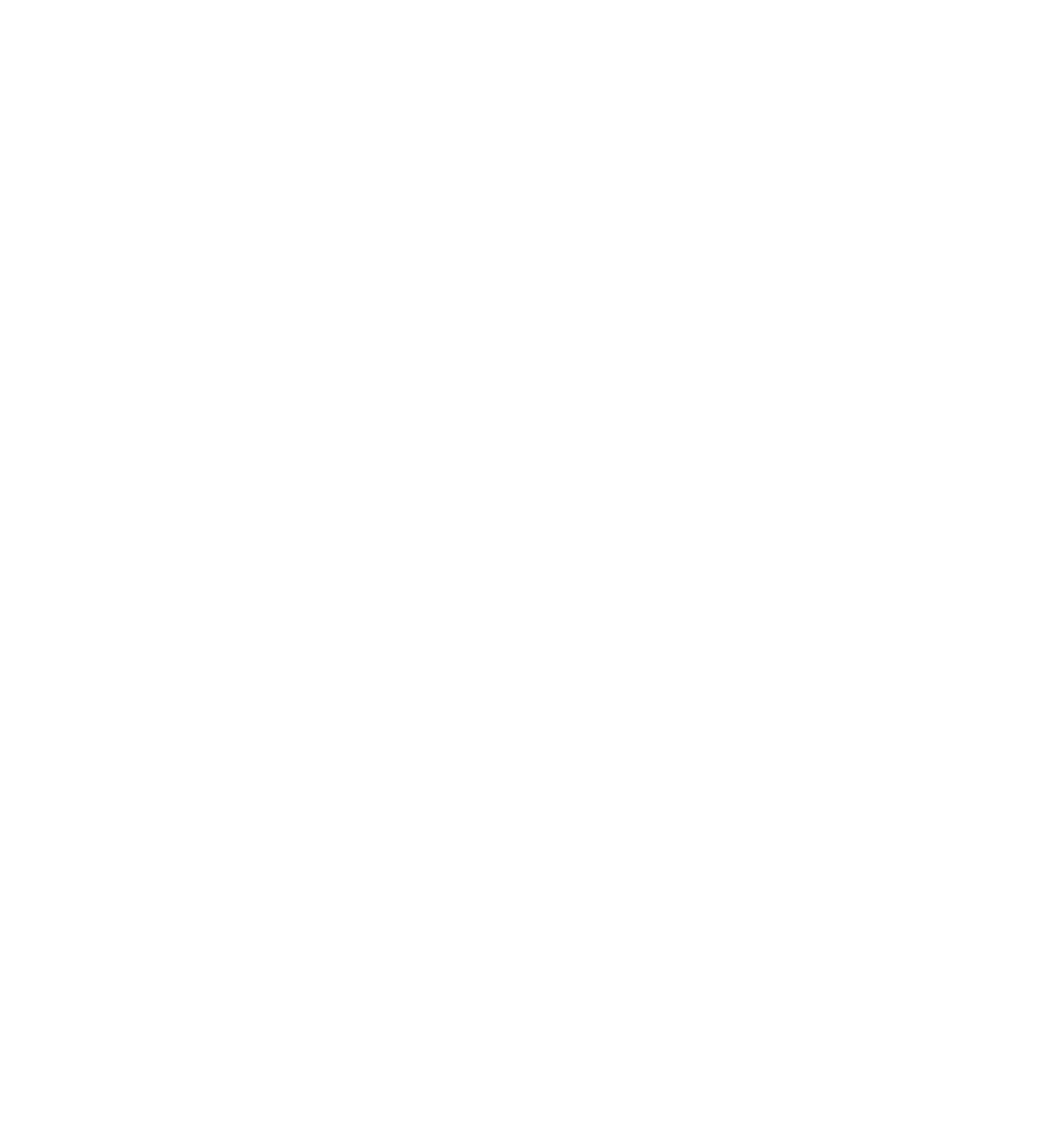
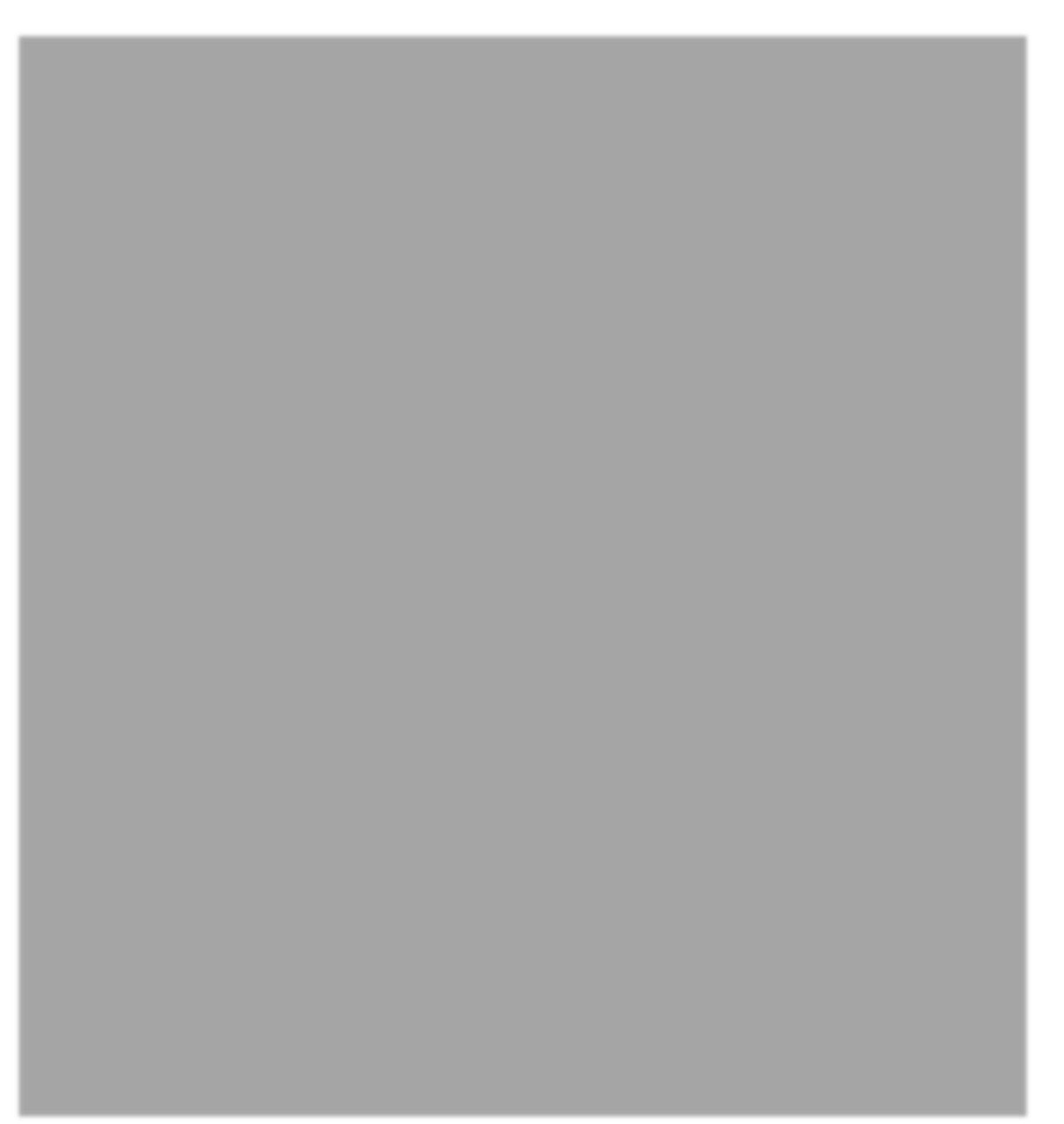
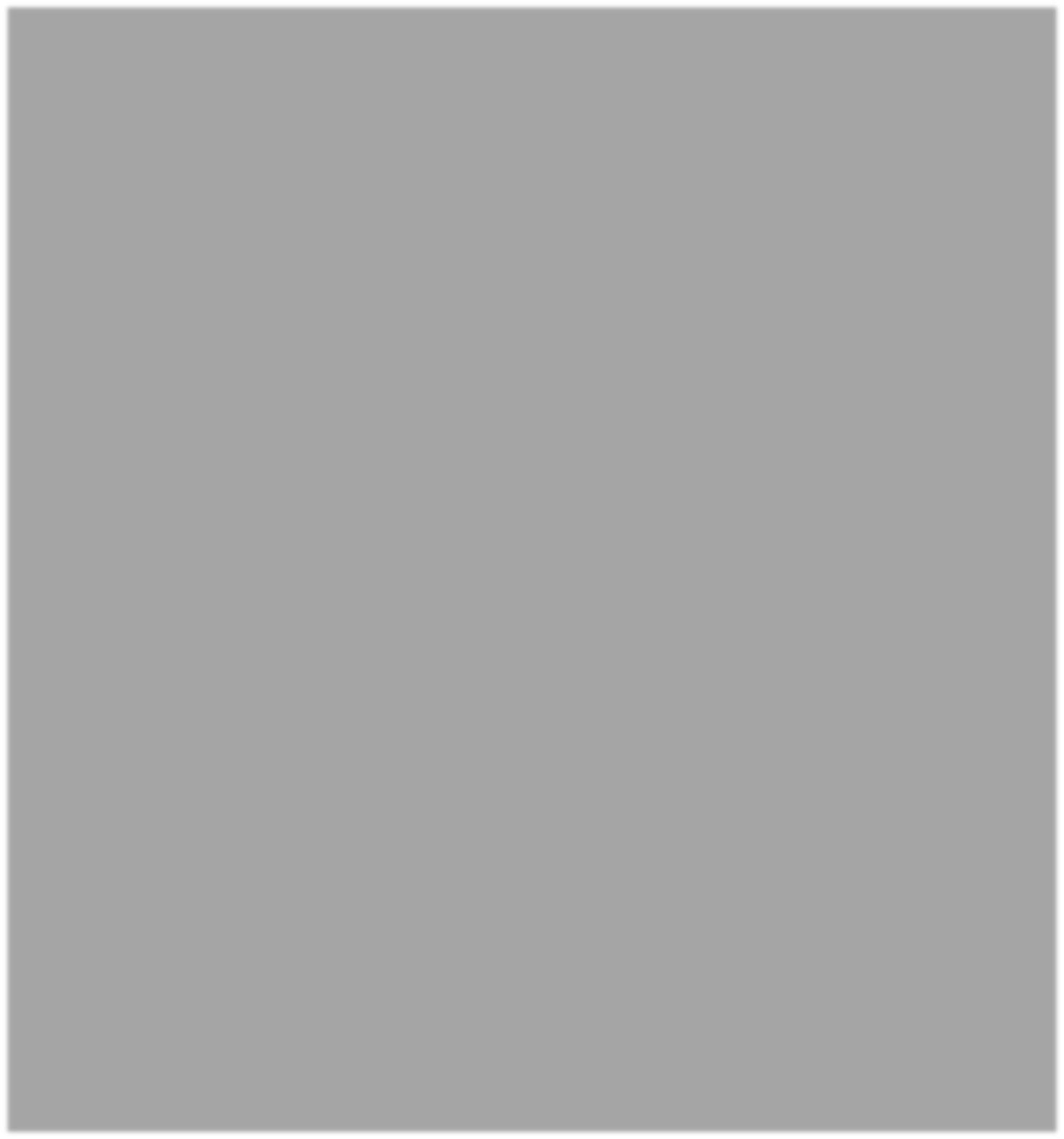
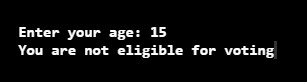
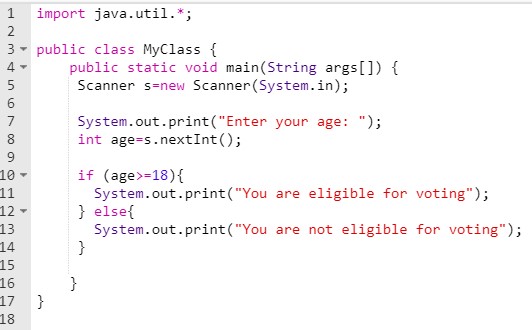
otherwise display “You are not elig

ible for voting”.

Sample Output 1:

Sample Output 2:

Solution:



Example 2:

Create a Java program that allows the user to accept three integers as input. Find the largest

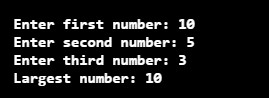
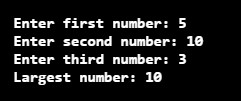
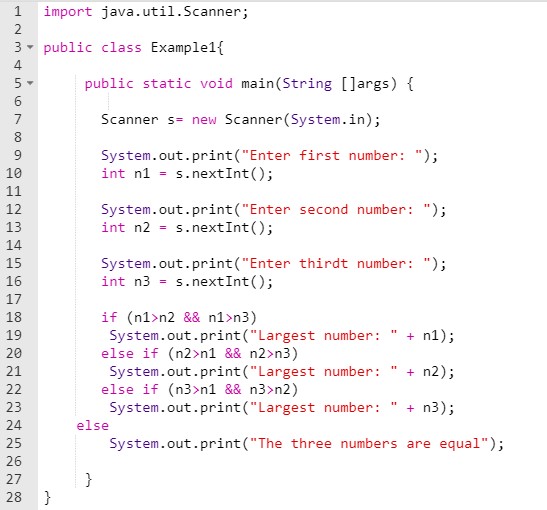
number.

Sample

Output 1:

Sample Output 2:

Solution:



2) Activity 3: Skill-building Activities (30 mins + 5 mins checking)

Test I. Fill in the blanks.

1. Print “Hello World” if x is greater than y.

int x=50; int y=10; \_if\_\_ (x \_>\_ y){

System.out.println(“Hello World”);

}

1. Print “Hello World” if x is equal to y.

int x=50; int y=50; \_if\_\_ (x \_= =\_ y){

System.out.println(“Hello World”);

}

1. Print “Yes” if x is equal to y otherwise print “No”

int x=50; int y=50;

\_if\_ (x \_= =\_ y){

System.out.println(“Yes”); } \_else\_ {

System.out.println(“No”);

}

|  |  |  |  |
| --- | --- | --- | --- |
| is equal to | y, print "2" if x | is greater than y | , otherwise print "3". |

1. Print "1" if x

int x=50;int y=50;

\_\_\_if\_\_\_(x\_==\_y){ System.out.println(“1”);

} \_else if\_ (x>y){

System.out.println(“2”);

}\_else\_ {

System.out.println(“3”);

}

1. Insert the missing parts to complete the following "short hand if” using ternary operator.

int time = 20;

String result = \_(\_time<18\_)\_ \_\_?\_\_ “Good day.” \_\_:\_\_ “Good evening”;

Test II. Multiple Choice. Encircle the letter that corresponds to the best answer.

1. How many choices are possible when using a single if – else statement?

* B

1. What does the following code fragment write to the monitor? int sum = 14; if ( sum < 20 )

System.out.print("Under ");

else

System.out.print("Over ");

System.out.print("the limit ");

* C

1. What does the following code fragment write to the monitor? int sum = 14; if ( sum < 20 )

System.out.print("Under ");

else{

System.out.print("Over ");

System.out.print("the limit ");

}

* D

1. What does the following code fragment write to the monitor?

int sum = 94; if ( sum < 20 ){

System.out.print("Under ");

System.out.print("the limit ");

}

else{

System.out.print("Over ");

System.out.print("the limit ");

}

* D

1. What does the following code fragment write to the monitor?

int sum = 7;

if ( sum > 20 )

{

System.out.print("You win ");

}

else

{

System.out.print("You lose ");

}

System.out.println("the prize.");

* D

1. What does the following code fragment write to the monitor?

int sum = 21; if ( sum == 20 ){ {

System.out.print("You win "); }

else{

System.out.print("You lose "); }

System.out.println("the prize.");

* D

1. What does the following code fragment write to the monitor?

int sum = 21; if ( sum != 20 )

System.out.print("You win ");

else

System.out.print("You lose ");

System.out.println("the prize.");

* D

1. A sequence of statements contained within a pair of braces ("{" and "}") is called a:

* A

1. Evaluate (to true or false) each of the following expressions:

14 <= 14 14 < 14 -9 > -25 -25 > -9

* D

1. Say that value has a 19 stored in it, and that extra has a 25 stored in it. Evaluate (to true or false) each of the following expressions:

value <= extra extra < value value > -25 value >= extra

* A

When you are done answering the self – assessment, please check your answers against the Key to Corrections found at the end of this Student Activity Sheets. Write your score on your paper.

1. Activity 4: What I Know Chart (3 mins)

Let us monitor how far you have learned in this lesson by reviewing the questions in the What I Know Chart from Activity 1. Please write your answers to the questions based on what you **now** know in the third column of the table.

|  |  |  |
| --- | --- | --- |
| What I Know | Questions: | What I Learned (Activity 4) |
|  | 1. What are the three types of control structures? | Sequential, Selection & Repetition |
|  | 2. What statement will be used to specify that a block of code will be executed if the condition is true? | do-while loop |
|  | 3. What method from the Utility package will be used to read user – input? | nextInt() |

3.) Activity 5: Check for Understanding (77 mins including hands – on + 10 mins checking)

Summative Test: Programming. You are going to solve the following problems using the Java programming language. Write your solution on the space provided.

1. Write a Java program that takes hours and minutes as input, and calculates the total number of minutes.

// Java program to convert hours

// into minutes and seconds

Import java.io.\*;

Class GFG {

// Function to convert hours

// into minutes and seconds

Static void ConvertHours(int n)

{

Int minutes, seconds;

Minutes = n \* 60;

Seconds = n \* 3600;

System.out.println( “Minutes = “ + minutes

+ “, Seconds = “ + seconds);

}

// Driver code

Public static void main (String[] args) {

Int n = 5;

ConvertHours(n);

}

}

1. Write a Java program to check whether a triangle is equilateral, Isosceles or scalene. Input the three sides of the triangle.

// Function to check if the triangle

// is equilateral or isosceles or scalene

Static void checkTriangle(int x, int y, int z)

{

// Check for equilateral triangle

If (x == y && y == z )

System.out.println(“Equilateral Triangle”);

// Check for isoceles triangle

Else if (x == y || y == z || z == x )

System.out.println(“Isoceles Triangle”);

// Otherwise scalene triangle

Else

System.out.println(“Scalene Triangle”);

}

// Driver Code

Public static void main(String[] args)

{

// Given sides of triangle

Int x = 8, y = 7, z = 9;

// Function call

checkTriangle(x, y, z);

}

}

# LESSON WRAP-UP

1) Activity 6: Thinking about Learning (5 mins)

1. Work Tracker

You are done with this session! Let’s track your progress. Shade the session number you just completed.



1. Think about your Learning

1. What particular part of the lesson that you find it easy?

* In this lesson I find that creating a flowchart is easier than doing it and run as a program.

1. In today’s lesson, what strategy did you use that turned out to be effective?

* The Strategy that I did is advance reading to this lesson.

**FAQs**

1. Do I need to use nextInt() always even If I will be reading floating point values?

No. The method that you are going to use would depend on the values that you are going to read. If you want to read floating point values, use nextFloat() or nextDouble().

1. Is it necessary to write a pair of { } under the if (condition) even if there is only one statement to be executed?

No. If there is only one statement to be executed, there’s no need to write a pair of { }. Otherwise, if you need to execute block of statements, then a pair of { } is needed

**KEY TO CORRECTIONS**

Activity 3

3) Activity 3: Skill-building Activities

Test I. Fill in the blanks.

1. Print “Hello World” if x is greater than y.

int x=50; int y=10; **if** (x **>** y){

System.out.println(“Hello World”);

}

1. Print “Hello World” if x is equal to y.

int x=50; int y=50; **if** (x **= =** y){

System.out.println(“Hello World”);

}

1. Print “Yes” if x is equal to y otherwise print “No”

int x=50; int y=50; **if** (x **= =** y){

System.out.println(“Yes”); } **else** {

System.out.println(“No”);

}

|  |  |  |  |
| --- | --- | --- | --- |
| is equal to | y, print "2" if x | is greater than y | , otherwise print "3". |

1. Print "1" if x

int x=50; int y=50; **if** (x **= =** y){

System.out.println(“1”);

} **else if** (x>y){

System.out.println(“2”);

} **else** {

System.out.println(“3”);

}

1. Insert the missing parts to complete the following "short hand if” using ternary operator.

int time = 20;

String result = **(**time<18 **) ?** “Good day.” **:** “Good evening”;

Test II. Multiple Choice. Encircle the letter that corresponds to the best answer.

1. How many choices are possible when using a single if – else statement?
   * 1. 1 c. 3
     2. **2**  d. 4

1. What does the following code fragment write to the monitor? int sum = 14; if ( sum < 20 )

System.out.print("Under ");

else

System.out.print("Over ");

System.out.print("the limit ");

* 1. Under **c. Under the limit**
  2. Over d. Over the limit

1. What does the following code fragment write to the monitor? int sum = 24; if ( sum < 20 )

System.out.print("Under ");

else{

System.out.print("Over ");

System.out.print("the limit ");

}

* 1. Under c. Under the limit
  2. Over **d. Over the limit**

1. What does the following code fragment write to the monitor?

int sum = 94; if ( sum < 20 ){

System.out.print("Under ");

System.out.print("the limit ");

}

else{

System.out.print("Over ");

System.out.print("the limit ");

}

* + 1. Under c. Under the limit
    2. Over **d. Over the limit**

1. What does the following code fragment write to the monitor?

int sum = 7;

if ( sum > 20 )

{

System.out.print("You win ");

}

else

{

System.out.print("You lose ");

}

System.out.println("the prize.");

* + 1. You win c. You win the prize
    2. You lose **d. You lose the prize.**

1. What does the following code fragment write to the monitor?

int sum = 21; if ( sum == 20 ){ {

System.out.print("You win "); }

else{

System.out.print("You lose "); }

System.out.println("the prize.");

* + 1. You win c. You win the prize
    2. You lose **d. You lose the prize.**

1. What does the following code fragment write to the monitor?

int sum = 21; if ( sum != 20 )

System.out.print("You win ");

else

System.out.print("You lose ");

System.out.println("the prize.");

* 1. You win c. You win the prize
  2. You lose **d. You lose the prize.**

1. A sequence of statements contained within a pair of braces ("{" and "}") is called a:

* 1. **block**  c. branch
  2. blob d. brick

1. Evaluate (to true or false) each of the following expressions:

14 <= 14 14 < 14 -9 > -25 -25 > -9

* 1. true true true true
  2. true false false false
  3. true false true true
  4. **true false true false**

1. Say that value has a 19 stored in it, and that extra has a 25 stored in it. Evaluate (to true or false) each of the following expressions:

value <= extra extra < value value > -25 value >= extra

* 1. **true false true false**
  2. true true true false
  3. false false true false
  4. false false true true

Note: There is no answer – key to summative test.