PRACTISE 5:

import java.util.Scanner;

public class ColorRange {

public static void main(String[] args) {

// Create a Scanner object for user input

Scanner scanner = new Scanner(System.in); // Define the valid range for each color component int minRange = 0; int maxRange = 255;

// Prompt user to enter RGB values

System.out.print(&quot;Enter the Red component (0-255): &quot;);

int red = scanner.nextInt();

System.out.print(&quot;Enter the Green component (0-255): &quot;); int green = scanner.nextInt();

System.out.print(&quot;Enter the Blue component (0-255): &quot;); int blue = scanner.nextInt();

// Check if the RGB values are within the specified range boolean isValidRed = red &gt;= minRange &amp;&amp; red &lt;= maxRange; boolean isValidGreen = green &gt;= minRange &amp;&amp; green &lt;= maxRange; boolean isValidBlue = blue &gt;= minRange &amp;&amp; blue &lt;= maxRange;

// Display results

System.out.println(&quot;\nColor Component Validity:&quot;);

System.out.println(&quot;Red: &quot; + (isValidRed ? &quot;Valid&quot; : &quot;Invalid&quot;));

System.out.println(&quot;Green: &quot; + (isValidGreen ? &quot;Valid&quot; : &quot;Invalid&quot;));

System.out.println(&quot;Blue: &quot; + (isValidBlue ? &quot;Valid&quot; : &quot;Invalid&quot;));

// Close the scanner scanner.close();

}

}

CODE:

import java.util.Scanner;

public class ColorRange {

public static void main(String[] args) {

// Create a Scanner object for user input

Scanner scanner = new Scanner(System.in);

// Define the valid range for each color component int minRange = 0; int maxRange = 255;

// Prompt user to enter RGB values

System.out.print("Enter the Red component (0-255): "); int red = scanner.nextInt();

System.out.print("Enter the Green component (0-255): "); int green = scanner.nextInt();

System.out.print("Enter the Blue component (0-255): "); int blue = scanner.nextInt();

// Check if the RGB values are within the specified range boolean isValidRed = red >= minRange && red <= maxRange; boolean isValidGreen = green >= minRange && green <= maxRange; boolean isValidBlue = blue >= minRange && blue <= maxRange;

// Display results

System.out.println("\nColor Component Validity:"); System.out.println("Red: " + (isValidRed ? "Valid" : "Invalid"));

System.out.println("Green: " + (isValidGreen ? "Valid" : "Invalid"));

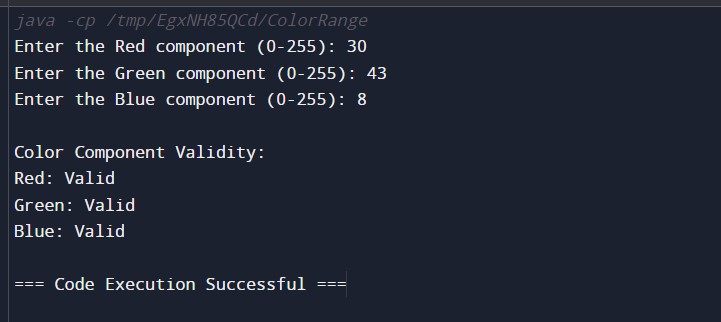
System.out.println("Blue: " + (isValidBlue ? "Valid" : "Invalid"));

// Close the scanner scanner.close();

}

}

OUTPUT:



2. To build a TrafficLightChecker class, you should focuon creating a system that simulates the behavior of traffic lights. This simulation can be expanded to include functionalities such as checking the current light status, determining the duration of each light phase, and providing a mechanism for switching between lights.

import java.util.Scanner; public class TrafficLightChecker {

// Enum to define traffic light states

private enum TrafficLight {

RED, YELLOW, GREEN

}

// Method to get the next traffic light based on current light private static TrafficLight getNextLight(TrafficLight current) { switch (current) { case RED:

return TrafficLight.GREEN; case YELLOW: return TrafficLight.RED;

case GREEN:

return TrafficLight.YELLOW; default:

throw new IllegalArgumentException(&quot;Unexpected value: &quot; + current);

}

}

// Method to display the traffic light status private static void displayStatus(TrafficLight light) { switch (light) {

case RED:

System.out.println(&quot;The light is RED. Please stop.&quot;); break; case YELLOW:

System.out.println(&quot;The light is YELLOW. Prepare to stop.&quot;); break;

case GREEN:

System.out.println(&quot;The light is GREEN. You may go.&quot;);

break;

}

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Prompt user for the initial traffic light state

System.out.print(&quot;Enter the current traffic light color (RED,

YELLOW, GREEN): &quot;);

String input = scanner.next().toUpperCase(); TrafficLight currentLight;

try {

// Convert the input string to TrafficLight enum currentLight = TrafficLight.valueOf(input);

} catch (IllegalArgumentException e) {

System.out.println(&quot;Invalid color entered. Please enter RED, YELLOW, or GREEN.&quot;); scanner.close(); return;

}

// Display the current light status displayStatus(currentLight);

// Determine the next traffic light state

TrafficLight nextLight = getNextLight(currentLight);

// Display the next light status

System.out.println(&quot;The next light will be: &quot; + nextLight); displayStatus(nextLight);

// Close the scanner scanner.close();

}

}

CODE:

import java.util.Scanner;

public class TrafficLightChecker { private enum TrafficLight {

RED, YELLOW, GREEN

}

private static TrafficLight getNextLight(TrafficLight current) { switch (current) { case RED:

return TrafficLight.GREEN; case YELLOW:

return TrafficLight.RED; case GREEN:

return TrafficLight.YELLOW;

default:

throw new IllegalArgumentException("Unexpected value: " + current);

}

}

private static void displayStatus(TrafficLight light) { switch (light) { case RED:

System.out.println("The light is RED. Please stop.");

break;

case YELLOW:

System.out.println("The light is YELLOW. Prepare to stop.");

break;

case GREEN:

System.out.println("The light is GREEN. You may go.");

break;

}

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the current traffic light color (RED, YELLOW, GREEN): ");

String input = scanner.next().toUpperCase();

TrafficLight currentLight;

try {

currentLight = TrafficLight.valueOf(input);

} catch (IllegalArgumentException e) {

System.out.println("Invalid color entered. Please enter RED, YELLOW, or GREEN.");

scanner.close(); return;

}

displayStatus(currentLight);

TrafficLight nextLight = getNextLight(currentLight);

System.out.println("The next light will be: " + nextLight);

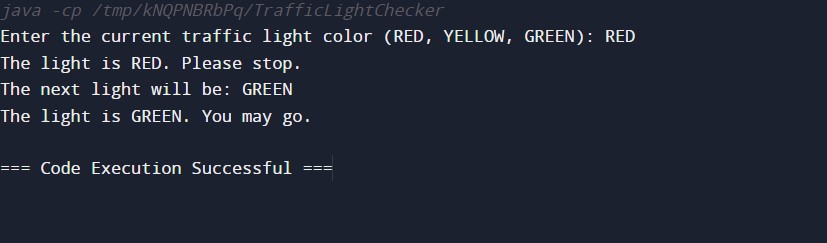
displayStatus(nextLight);

scanner.close();

}

}

OUTPUT:



3. To implement a TrafficLightSwitch class that simulate switching traffic lights, you might want to create functionality for managing the current state of the traffic light, switching between states, and possibly displaying information about the light. Below is a detailed example of how you could set up this class:

import java.util.Scanner; public class TrafficLightSwitch { // Enum to define traffic light states private enum TrafficLight {

RED, YELLOW, GREEN

}

// Method to get the next traffic light based on current light private static TrafficLight getNextLight(TrafficLight current) { switch (current) { case RED:

return TrafficLight.GREEN;

case YELLOW: return TrafficLight.RED; case GREEN:

return TrafficLight.YELLOW; default:

throw new IllegalArgumentException(&quot;Unexpected value: &quot; + current);

}

}

// Method to display the traffic light status private static void displayStatus(TrafficLight light) { switch (light) { case RED:

System.out.println(&quot;The light is RED. Please stop.&quot;); break;

case YELLOW:

System.out.println(&quot;The light is YELLOW. Prepare to stop.&quot;); break; case GREEN:

System.out.println(&quot;The light is GREEN. You may go.&quot;); break;

}

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Prompt user for the initial traffic light state

System.out.print(&quot;Enter the current traffic light color (RED, YELLOW,

GREEN): &quot;);

String input = scanner.next().toUpperCase();

TrafficLight currentLight;

try {

// Convert the input string to TrafficLight enum currentLight = TrafficLight.valueOf(input);

} catch (IllegalArgumentException e) {

System.out.println(&quot;Invalid color entered. Please enter RED, YELLOW, or GREEN.&quot;);

scanner.close(); return;

}

// Display the current light status displayStatus(currentLight);

// Determine the next traffic light state

TrafficLight nextLight = getNextLight(currentLight);

// Display the next light status

System.out.println(&quot;The next light will be: &quot; + nextLight); displayStatus(nextLight);

// Close the scanner

scanner.close();

}

}

CORRECTED CODE:

import java.util.Scanner;

public class TrafficLightSwitch { private enum TrafficLight { RED, YELLOW, GREEN

}

private static TrafficLight getNextLight(TrafficLight current) { switch (current) { case RED:

return TrafficLight.GREEN; case YELLOW:

return TrafficLight.RED; case GREEN:

return TrafficLight.YELLOW; default:

throw new IllegalArgumentException("Unexpected value: " + current);

}

}

private static void displayStatus(TrafficLight light) {

switch (light) { case RED:

System.out.println("The light is RED. Please stop.");

break; case YELLOW:

System.out.println("The light is YELLOW. Prepare to stop.");

break;

case GREEN:

System.out.println("The light is GREEN. You may go.");

break;

}

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the current traffic light color (RED, YELLOW, GREEN): ");

String input = scanner.next().toUpperCase();

TrafficLight currentLight;

try {

currentLight = TrafficLight.valueOf(input);

} catch (IllegalArgumentException e) {

System.out.println("Invalid color entered. Please enter RED, YELLOW, or GREEN.");

scanner.close(); return;

}

displayStatus(currentLight);

TrafficLight nextLight = getNextLight(currentLight);

System.out.println("The next light will be: " + nextLight); displayStatus(nextLight);

scanner.close();

}

}

OUTPUT:

