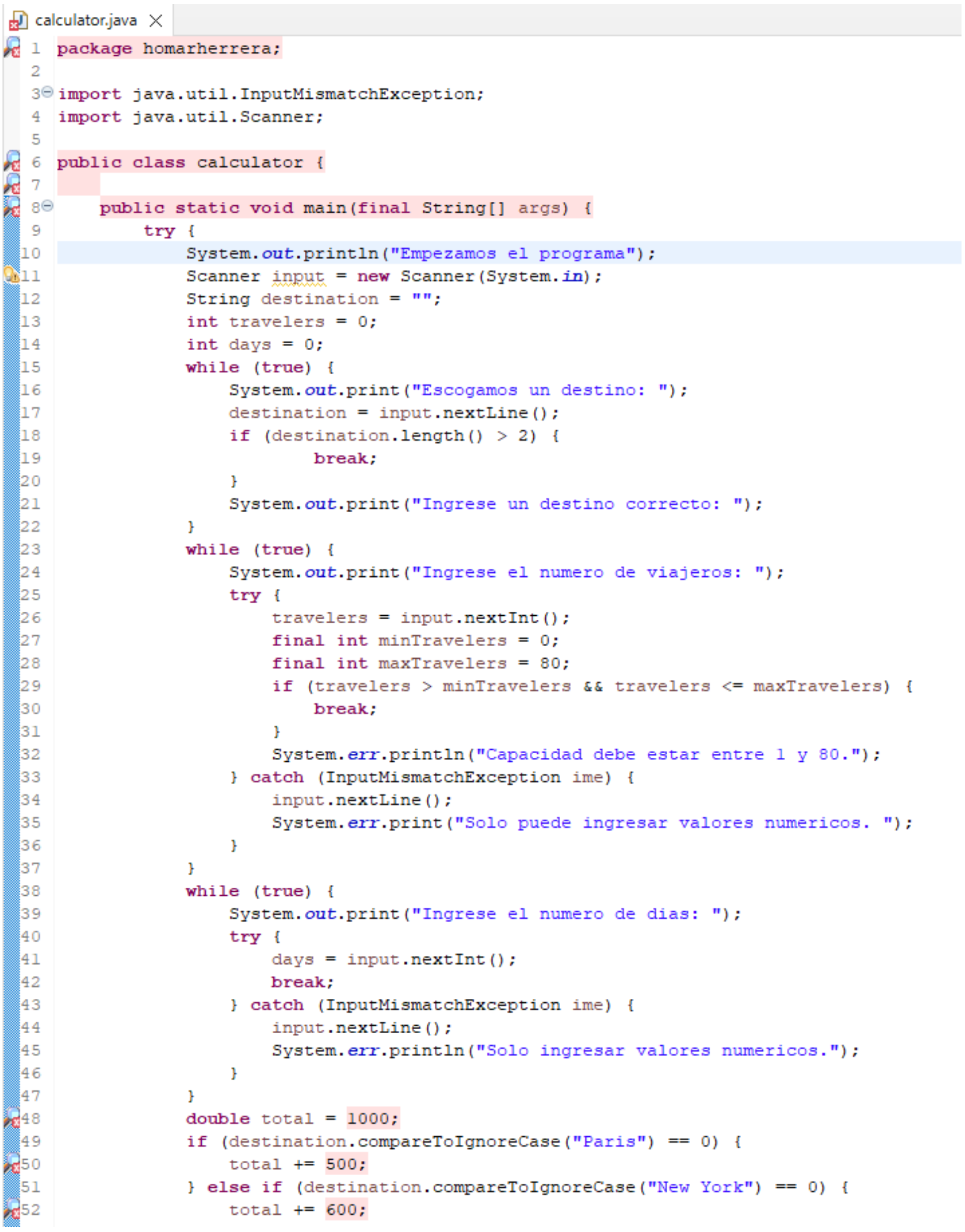
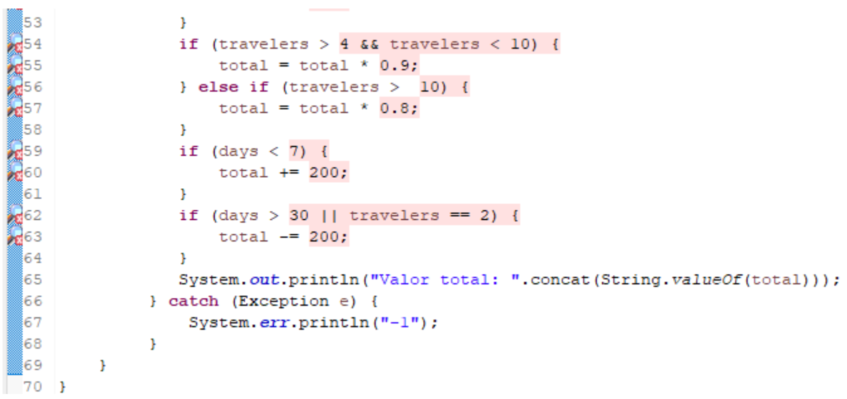
Original code where errors can be seen.

To carry out the activity, it was decided to create a linear code structure, which is attached in the following images.



```java
package homarherrera;

import java.util.InputMismatchException;
import java.util.Scanner;

public class calculator {

    public static void main(final String[] args) {
        try {
            System.out.println("Empezamos el programa");
            Scanner input = new Scanner(System.in);
            String destination = "";
            int travelers = 0;
            int days = 0;
            while (true) {
                System.out.print("Escogamos un destino: ");
                destination = input.nextLine();
                if (destination.length() > 2) {
                    break;
                }
                System.out.print("Ingrese un destino correcto: ");
            }
            while (true) {
                System.out.print("Ingrese el numero de viajeros: ");
                try {
                    travelers = input.nextInt();
                    final int minTravelers = 0;
                    final int maxTravelers = 80;
                    if (travelers > minTravelers && travelers <= maxTravelers) {
                        break;
                    }
                    System.err.println("Capacidad debe estar entre 1 y 80.");
                } catch (InputMismatchException ime) {
                    input.nextLine();
                    System.err.print("Solo puede ingresar valores numericos. ");
                }
            }
            while (true) {
                System.out.print("Ingrese el numero de dias: ");
                try {
                    days = input.nextInt();
                    break;
                } catch (InputMismatchException ime) {
                    input.nextLine();
                    System.err.println("Solo ingresar valores numericos.");
                }
            }
            double total = 1000;
            if (destination.compareToIgnoreCase("Paris") == 0) {
                total += 500;
            } else if (destination.compareToIgnoreCase("New York") == 0) {
                total += 600;
```

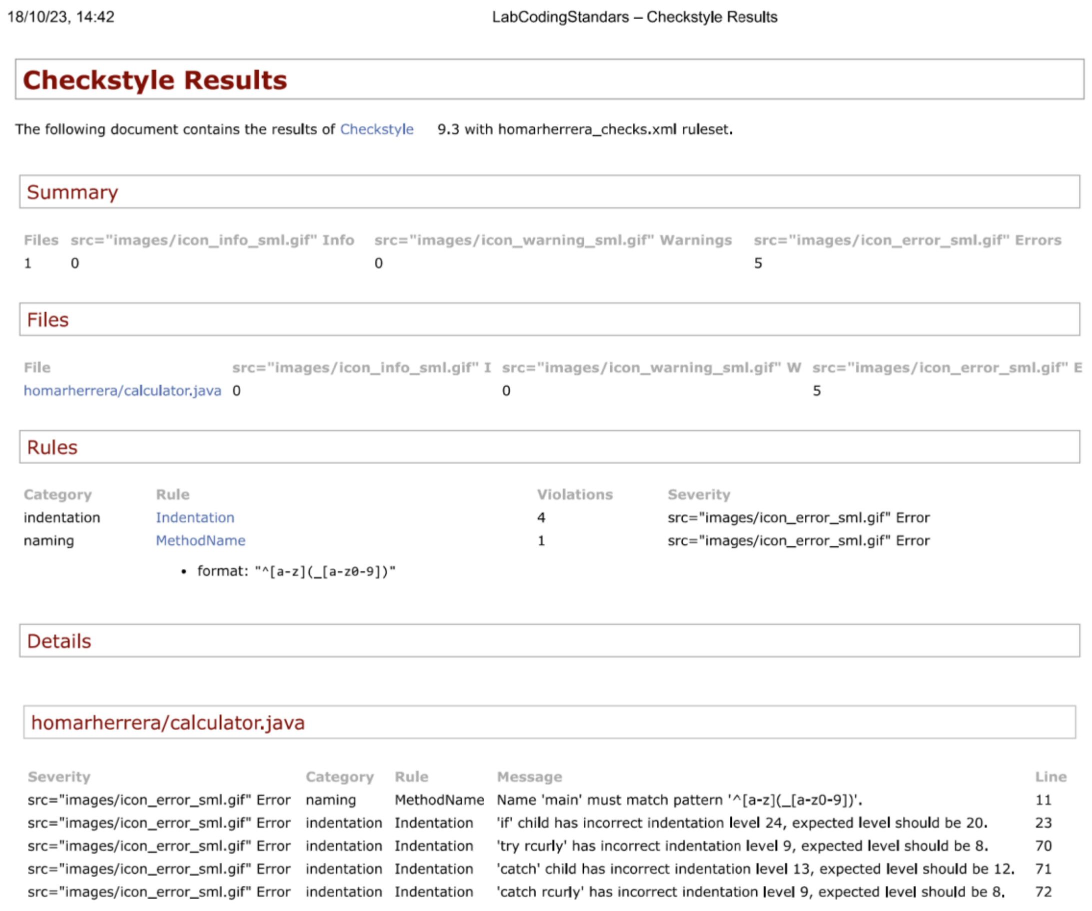


```java
            }
            if (travelers > 4 && travelers < 10) {
                total = total * 0.9;
            } else if (travelers >  10) {
                total = total * 0.8;
            }
            if (days < 7) {
                total += 200;
            }
            if (days > 30 || travelers == 2) {
                total -= 200;
            }
            System.out.println("Valor total: ".concat(String.valueOf(total)));
        } catch (Exception e) {
            System.err.println("-1");
        }
    }
}
```

With the installation of the Checkstyle plugin, the following rules were configured:

- Indentation.
- Java methods.
- Whitespace After.
- Missing javadoc method.
- Avoid static imports.

With the rules defined, the check was carried out using Maven, obtaining the following errors



18/10/23, 14:42 LabCodingStandars – Checkstyle Results

Checkstyle Results

The following document contains the results of Checkstyle 9.3 with homarherrera_checks.xml ruleset.

Summary

Files	src="images/icon_info_sml.gif" Info	src="images/icon_warning_sml.gif" Warnings	src="images/icon_error_sml.gif" Errors
1	0	0	5

Files

File	src="images/icon_info_sml.gif" I	src="images/icon_warning_sml.gif" W	src="images/icon_error_sml.gif" E
homarherrera/calculator.java	0	0	5

Rules

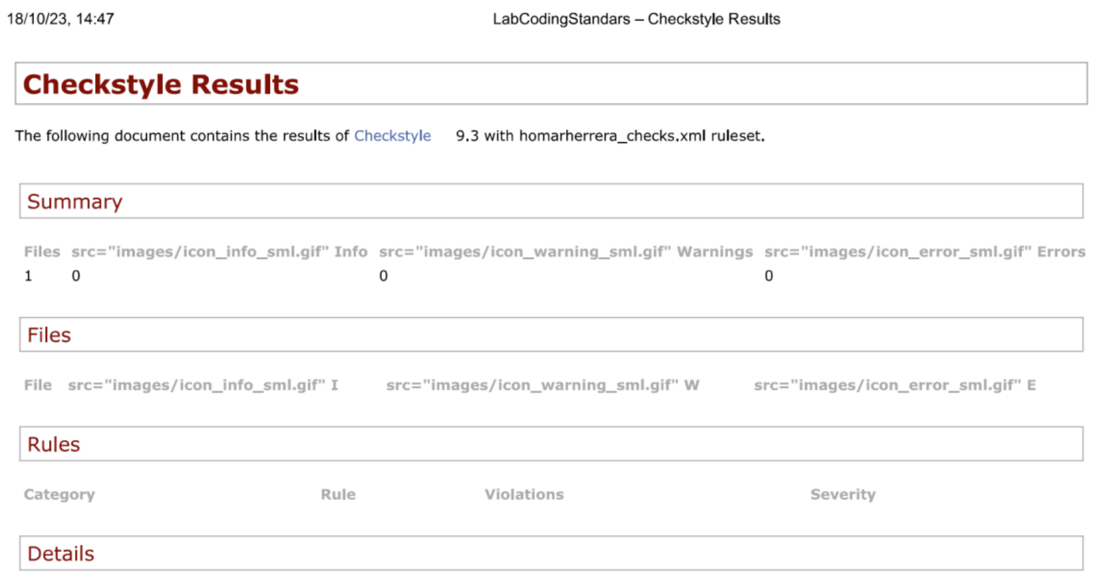
Category	Rule	Violations	Severity
indentation	Indentation	4	src="images/icon_error_sml.gif" Error
naming	MethodName • format: "^[a-z](_[a-z0-9])"	1	src="images/icon_error_sml.gif" Error

Details

homarherrera/calculator.java

Severity	Category	Rule	Message	Line
src="images/icon_error_sml.gif" Error	naming	MethodName	Name 'main' must match pattern '^[a-z](_[a-z0-9])'.	11
src="images/icon_error_sml.gif" Error	indentation	Indentation	'if' child has incorrect indentation level 24, expected level should be 20.	23
src="images/icon_error_sml.gif" Error	indentation	Indentation	'try rcurly' has incorrect indentation level 9, expected level should be 8.	70
src="images/icon_error_sml.gif" Error	indentation	Indentation	'catch' child has incorrect indentation level 13, expected level should be 12.	71
src="images/icon_error_sml.gif" Error	indentation	Indentation	'catch rcurly' has incorrect indentation level 9, expected level should be 8.	72

After reviewing the errors and doing some research into their solution, they were eliminated, resulting in the result when running Maven.



18/10/23, 14:47 LabCodingStandars – Checkstyle Results

Checkstyle Results

The following document contains the results of Checkstyle 9.3 with homarherrera_checks.xml ruleset.

Summary

Files	src="images/icon_info_sml.gif" Info	src="images/icon_warning_sml.gif" Warnings	src="images/icon_error_sml.gif" Errors
1	0	0	0

Files

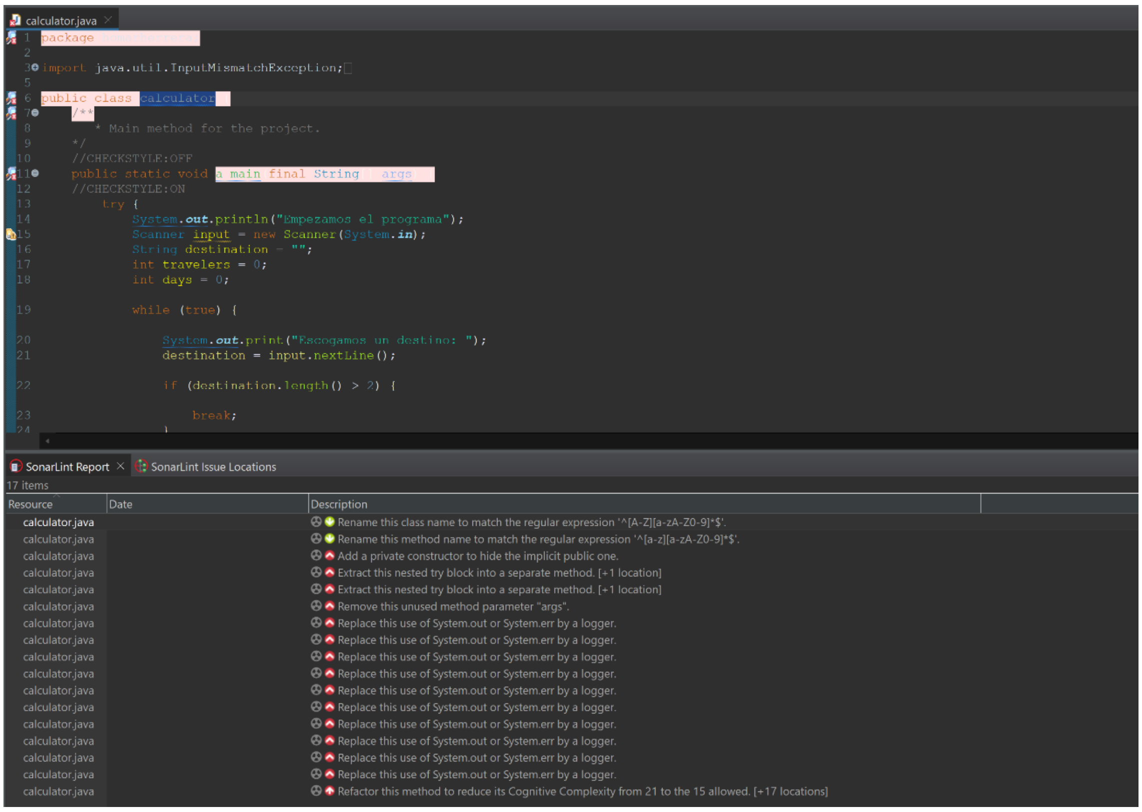
File	src="images/icon_info_sml.gif" I	src="images/icon_warning_sml.gif" W	src="images/icon_error_sml.gif" E

Rules

Category	Rule	Violations	Severity

Details

For the competition, the SonarLint tool was used because the program was written in Java and mainly complexity, variable declaration and recurrence errors were obtained.



calculator.java

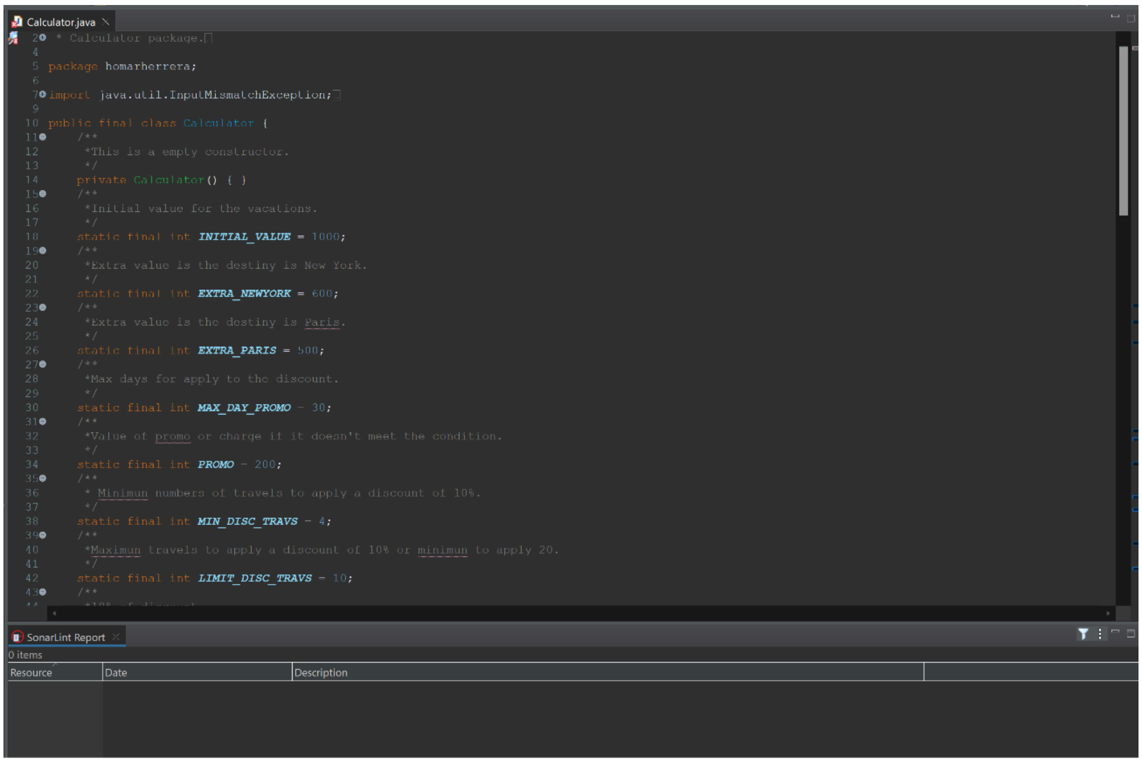
```java
package homarherrera;

import java.util.InputMismatchException;

public class calculator {
    /**
        * Main method for the project.
    */
    //CHECKSTYLE:OFF
    public static void a main(final String[] args) {
    //CHECKSTYLE:ON
        try {
            System.out.println("Empezamos el programa");
            Scanner input = new Scanner(System.in);
            String destination = "";
            int travelers = 0;
            int days = 0;

            while (true) {

                System.out.print("Escogamos un destino: ");
                destination = input.nextLine();

                if (destination.length() > 2) {

                    break;
                }
```

SonarLint Report | SonarLint Issue Locations

17 items

Resource	Date	Description
calculator.java		Rename this class name to match the regular expression '^[A-Z][a-zA-Z0-9]*$'.
calculator.java		Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
calculator.java		Add a private constructor to hide the implicit public one.
calculator.java		Extract this nested try block into a separate method. [+1 location]
calculator.java		Extract this nested try block into a separate method. [+1 location]
calculator.java		Remove this unused method parameter "args".
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Replace this use of System.out or System.err by a logger.
calculator.java		Refactor this method to reduce its Cognitive Complexity from 21 to the 15 allowed. [+17 locations]

After correcting the errors and investigating how to eliminate the complexity, we proceeded to eliminate the rule of replacing with a logger, obtaining the following report from SonarLint.



Calculator.java

```java
 * Calculator package.

package homarherrera;

import java.util.InputMismatchException;

public final class Calculator {
    /**
     *This is a empty constructor.
     */
    private Calculator() { }
    /**
     *Initial value for the vacations.
     */
    static final int INITIAL_VALUE = 1000;
    /**
     *Extra value is the destiny is New York.
     */
    static final int EXTRA_NEWYORK = 600;
    /**
     *Extra value is the destiny is Paris.
     */
    static final int EXTRA_PARIS = 500;
    /**
     *Max days for apply to the discount.
     */
    static final int MAX_DAY_PROMO = 30;
    /**
     *Value of promo or charge if it doesn't meet the condition.
     */
    static final int PROMO = 200;
    /**
     * Minimun numbers of travels to apply a discount of 10%.
     */
    static final int MIN_DISC_TRAVS = 4;
    /**
     *Maximun travels to apply a discount of 10% or minimun to apply 20.
     */
    static final int LIMIT_DISC_TRAVS = 10;
    /**
```

SonarLint Report

0 items

Resource	Date	Description