

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING
SOFTWARE ENGINEERING II
CODING STANDARDS - 2023

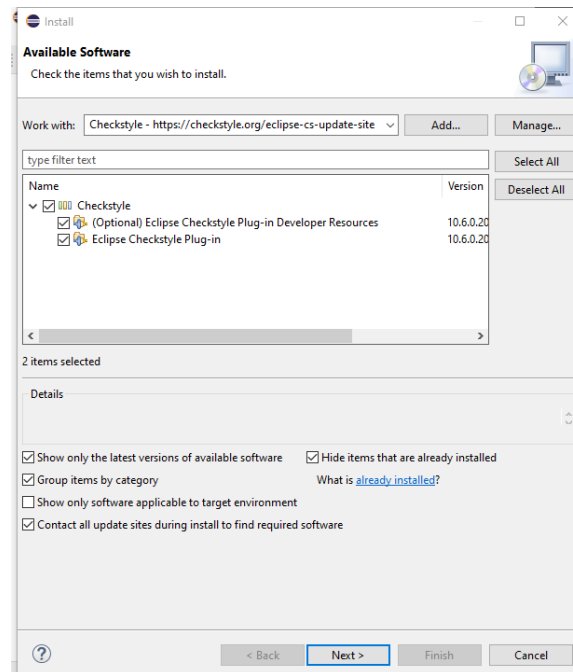
Name: María Alexandra Ramos Plúas

Reporte taller#1

Part 1: Install Maven for Eclipse.

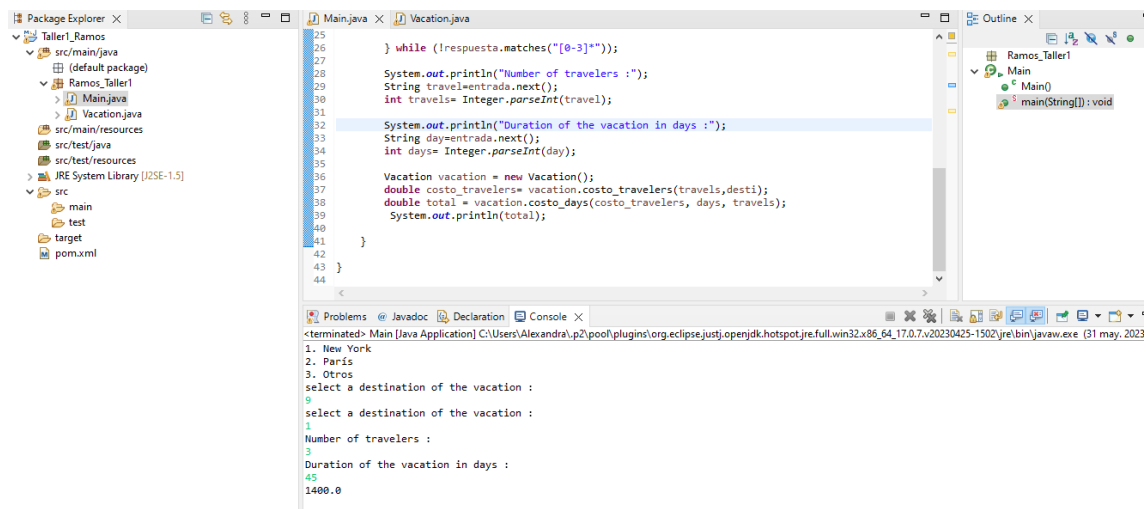
Maven is already installed in the new versions

Part 2: Install the Eclipse Checkstyle plug-in



Part 3: Create and configure the Project

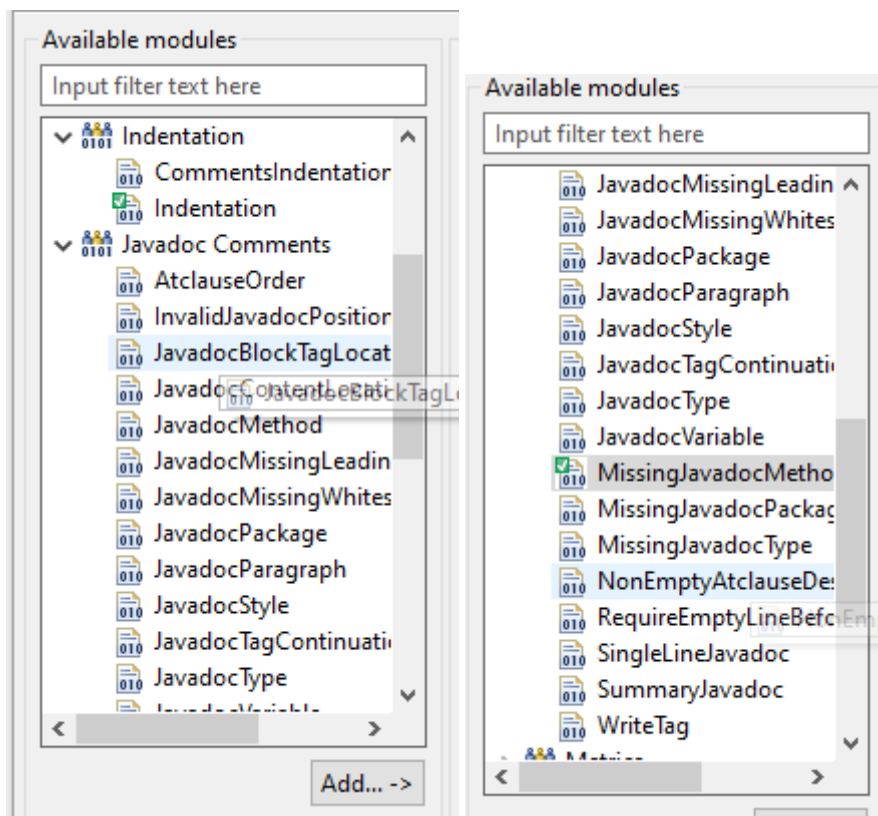
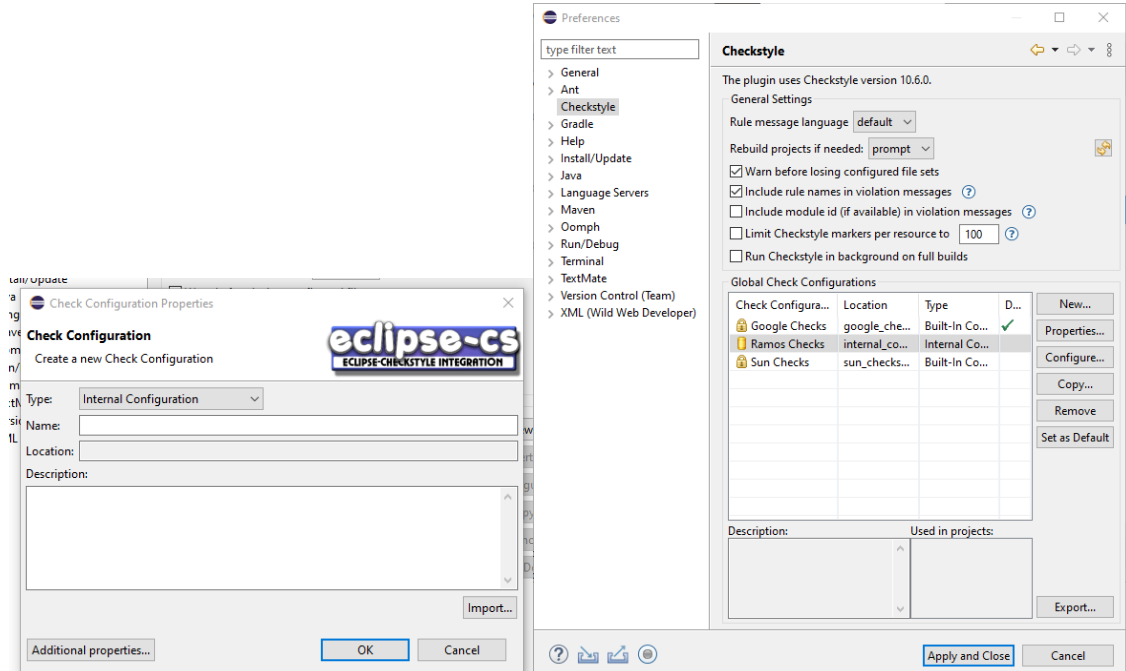
Project created and Checkstyle enabled



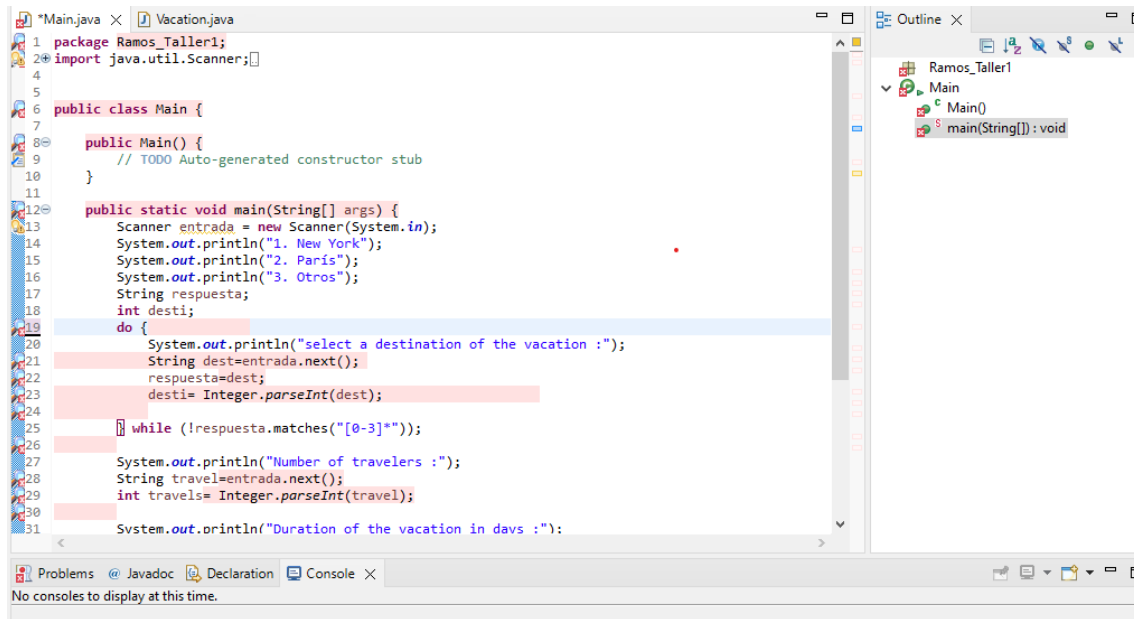
ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING
SOFTWARE ENGINEERING II
CODING STANDARDS - 2023

Part 4: Configure Checkstyle

Once the Checkstyle configuration is finished, the lines of code with any error are highlighted in pink color and if we place the cursor over them, it tells us the error.



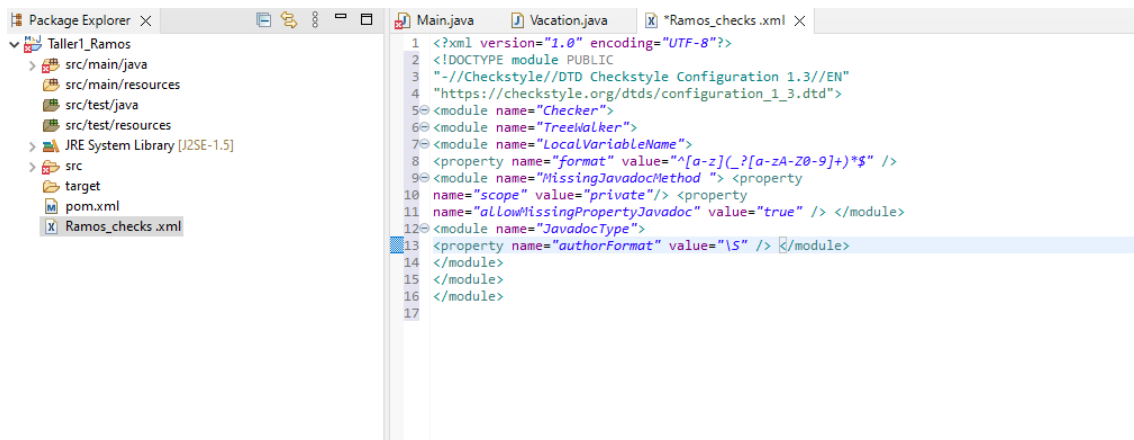
ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING
SOFTWARE ENGINEERING II
CODING STANDARDS - 2023



```
1 package Ramos_Taller1;
2 import java.util.Scanner;
3
4
5
6 public class Main {
7
8     public Main() {
9         // TODO Auto-generated constructor stub
10    }
11
12    public static void main(String[] args) {
13        Scanner entrada = new Scanner(System.in);
14        System.out.println("1. New York");
15        System.out.println("2. Paris");
16        System.out.println("3. Otros");
17        String respuesta;
18        int desti;
19        do {
20            System.out.println("select a destination of the vacation :");
21            String dest=entrada.next();
22            respuesta=dest;
23            desti= Integer.parseInt(dest);
24
25            while (!respuesta.matches("[0-3]*"));
26
27            System.out.println("Number of travelers :");
28            String travel=entrada.next();
29            int travels= Integer.parseInt(travel);
30
31            System.out.println("Duration of the vacation in days :");
```

Part 5: Configure Checkstyle using the XML

The Configure Checkstyle was made using the XML and an XML reader and tools had to be installed.

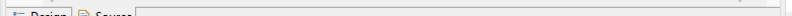


```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE module PUBLIC
3     "-//Checkstyle//DTD Checkstyle Configuration 1.3//EN"
4     "https://checkstyle.org/dtds/configuration_1_3.dtd">
5
6 <module name="Checker">
7     <module name="TreeWalker">
8         <module name="LocalVariableName">
9             <property name="format" value="^[a-z](_?[a-zA-Z0-9-9])*$" />
10        <module name="MissingJavadocMethod"> <property
11            name="scope" value="private"/> <property
12            name="allowMissingPropertyJavadoc" value="true" /> </module>
13        <module name="JavadocType">
14            <property name="authorFormat" value="\S" /> </module>
15        </module>
16    </module>
17 </module>
```

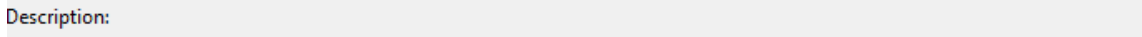
Part 6: Check for headers.

Let's add a rule to verify a fixed header

CODING STANDARDS - 2023



Part 7: Suppressing Checkstyle Tests



— 1 —

Repository:

https://github.com/maraplua/Coding_Standards_SWII