

**Homework Assignment 10 – Understanding and Using Static Code Analysis Tools**

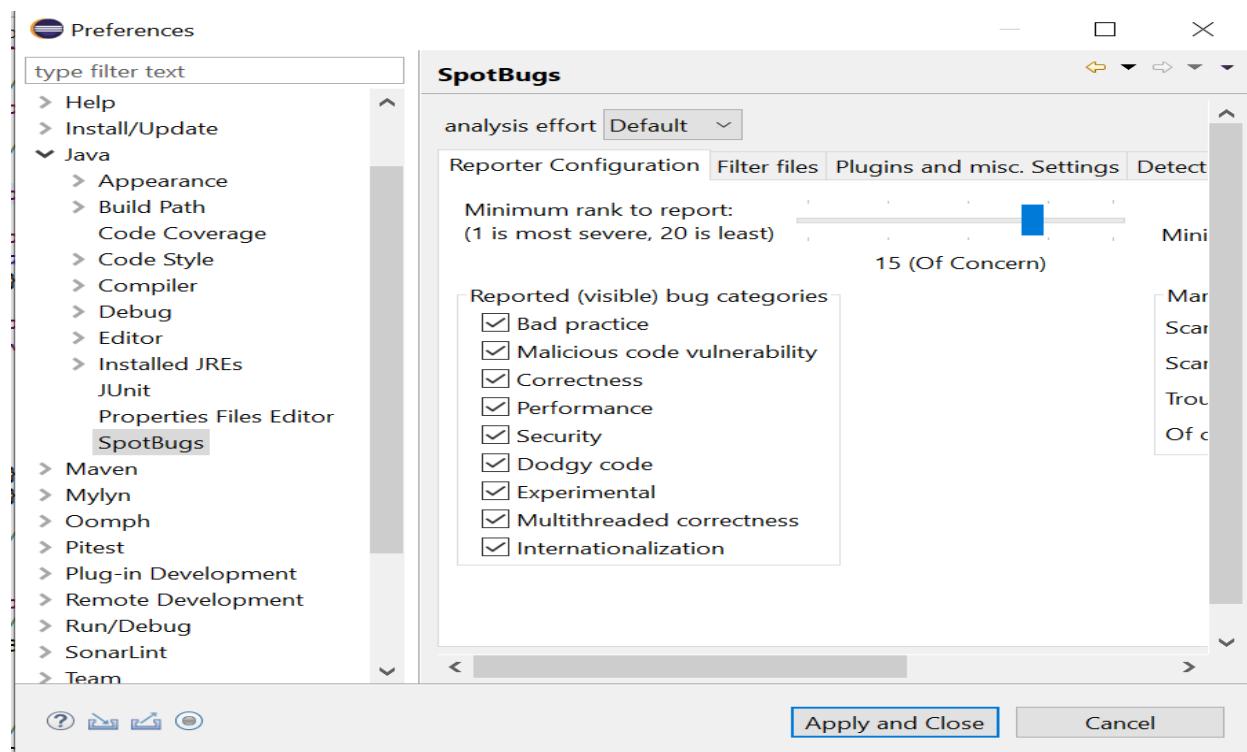
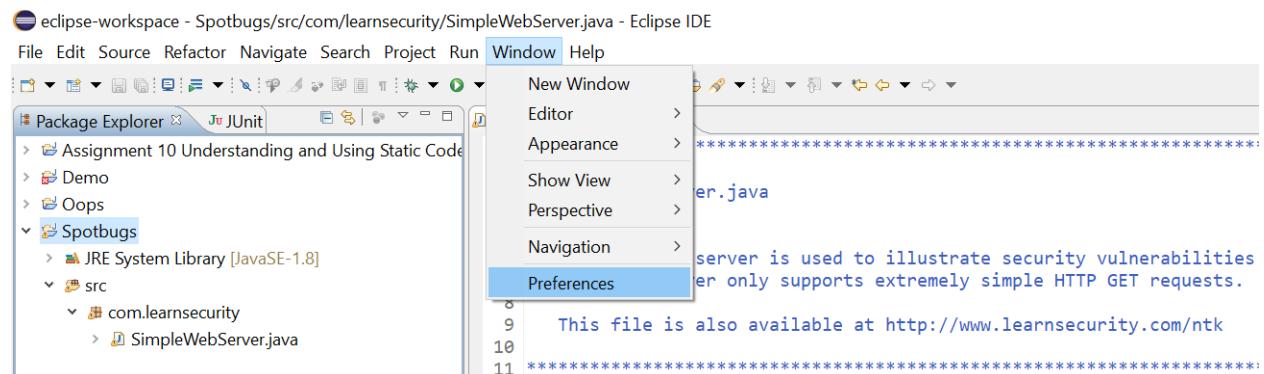
**Results in a separate file**

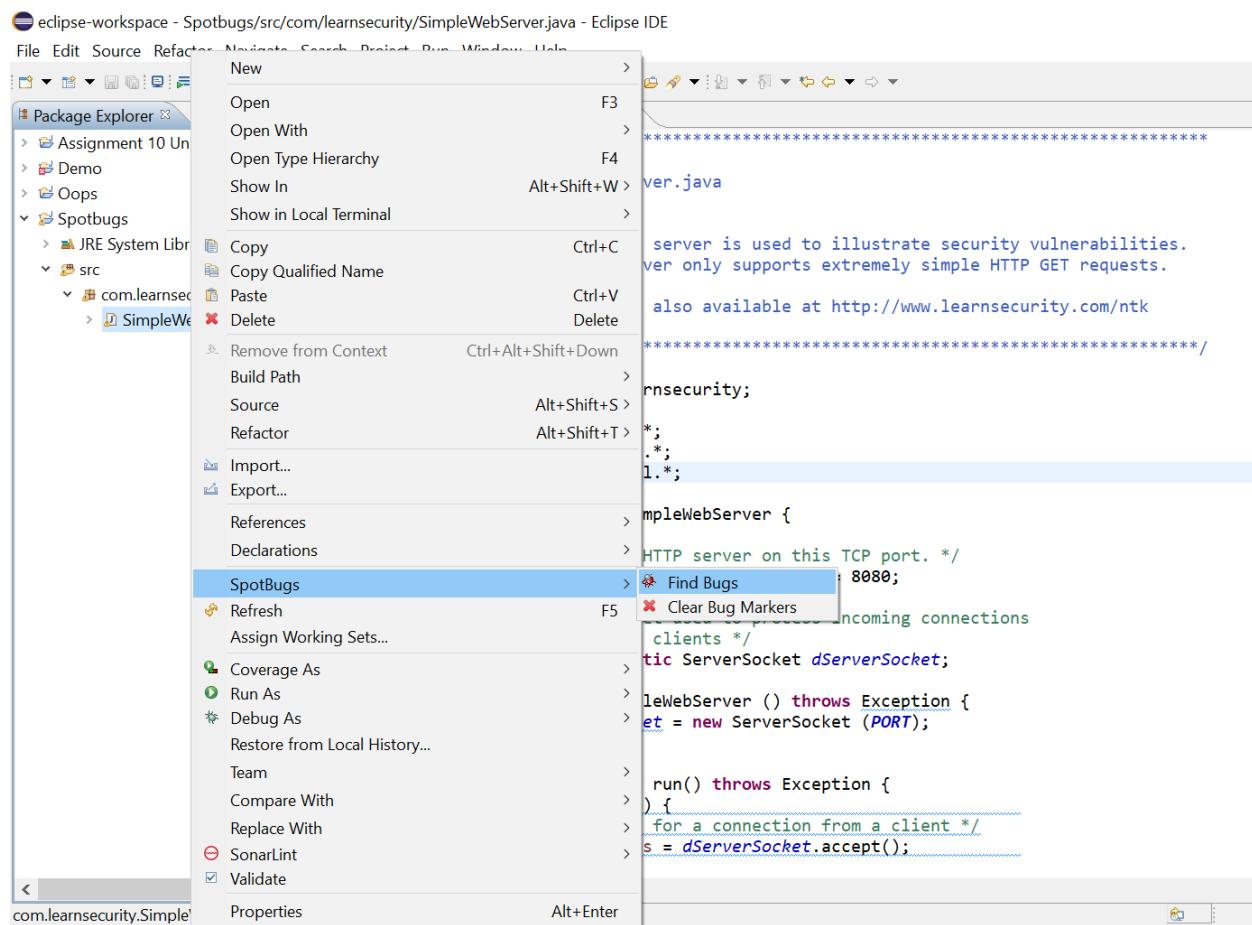
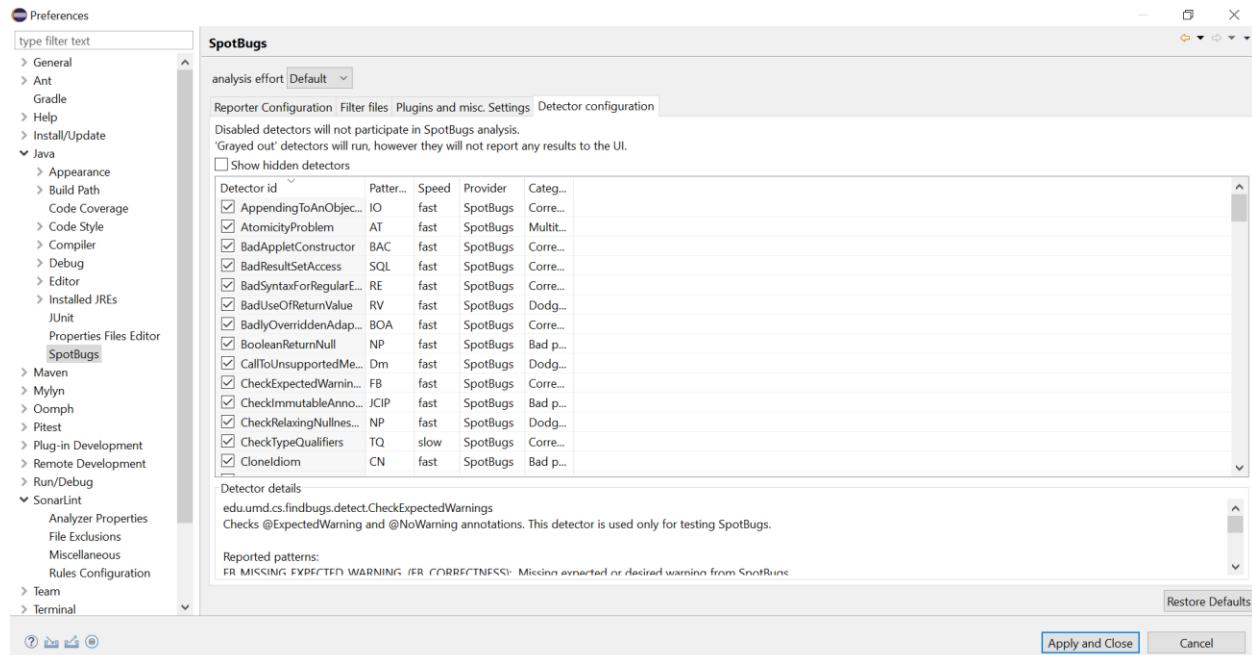
Name	UTA ID
Goutami Padmanabhan	1001669338

**a) Part 1 - SpotBugs**

Security rules enabled (along with required security plugins):

Window => Preferences





## Raw results provided (in separate file, directly output from tool):

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java

```
23  /* The socket used to process incoming connections
24  * from web clients */
25  private static ServerSocket dServerSocket;
26
27  public SimpleWebServer () throws Exception {
28      dServerSocket = new ServerSocket (PORT);
29  }
30
31  public void run() throws Exception {
32      while (true) {
33          /* wait for a connection from a client */
34          Socket s = dServerSocket.accept();
35
36          /* then process the client's request */
37          processRequest(s);
38      }
39  }
40
41  /* Reads the HTTP request from the client, and
42  * responds with the file the user requested or
43  * a HTTP error code. */
44  public void processRequest(Socket s) throws Exception {
45      /* used to read data from the client */
46      BufferedReader br =
47          new BufferedReader (
48              new InputStreamReader (s.getInputStream()));
49
50      /* used to write data to the client */
51      OutputStreamWriter osw =
52          new OutputStreamWriter (s.getOutputStream());
53
54      /* read the HTTP request from the client */
55      String request = br.readLine();
56
57      String command = null;
```

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

```

public void processRequest(Socket s) throws Exception {
    /* used to read data from the client */
    BufferedReader br =
        new BufferedReader (
            new InputStreamReader (s.getInputStream()));

    /* used to write data to the client */
    OutputStreamWriter osw =
        new OutputStreamWriter (s.getOutputStream());

    /* read the HTTP request from the client */
    String request = br.readLine();

    String command = null;
    String pathname = null;

    /* parse the HTTP request */
    StringTokenizer st =
        new StringTokenizer (request, " ");

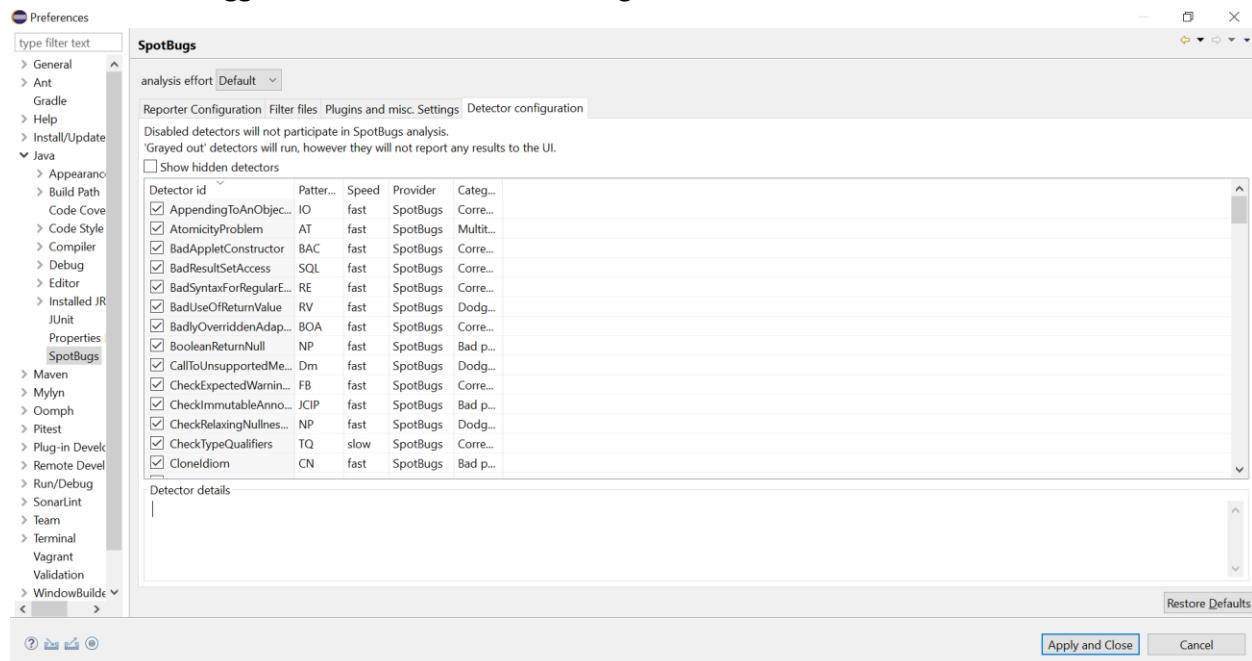
    command = st.nextToken();
    pathname = st.nextToken();

    if (command.equals("GET")) {
        /* if the request is a GET
           try to respond with the file
           the user is requesting */
        serveFile (osw,pathname);
    }
    else {
        /* if the request is a NOT a GET,
           return an error saying this server
           does not implement the requested command */
        osw.write ("HTTP/1.0 501 Not Implemented\n\n");
    }

    /* ----- THE FOLLOWING IS ALL ADDED ----- */
}

```

## Turned on most aggressive mode in tool for finding defects:



eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

- Assignment 10 Understanding and Using Static Code
- Demo
- Oops
- Spotbugs (2)
  - JRE System Library [JavaSE-1.8]
  - src (2)
    - com.learnsecurity (2)
    - SimpleWebServer.java (2)

SimpleWebServer.java

```

21  /* Run the HTTP server on this TCP port. */
22  private static final int PORT = 8080;
23
24  /* The socket used to process incoming connections
   from web clients */
25  private static ServerSocket dServerSocket;
26
27  public SimpleWebServer () throws Exception {
28      dServerSocket = new ServerSocket (PORT);
29  }
30
31
32  public void run() throws Exception {
33      while (true) {
34          /* wait for a connection from a client */
35      }
36  }

```

Bug Info

SimpleWebServer.java: 29

Navigation

Write to static field com.learnsecurity.SimpleWebServer.dServerSocket from instance method new com.learnsecurity.SimpleWebServer()

Field com.learnsecurity.SimpleWebServer.dServerSocket

**Bug:** Write to static field com.learnsecurity.SimpleWebServer.dServerSocket from instance method new com.learnsecurity.SimpleWebServer()

This instance method writes to a static field. This is tricky to get correct if multiple instances are being manipulated, and generally bad practice.

**Rank:** Of Concern (15), **confidence:** High  
**Pattern:** ST\_WRITE\_TO\_STATIC\_FROM\_INSTANCE\_METHOD  
**Type:** ST, **Category:** STYLE (Dodgy code)

Problems Javadoc Declaration Bug Explorer

Spotbugs (2)
 

- Of Concern (2)
  - High confidence (1)
    - Write to static field from instance method (1)
      - Write to static field com.learnsecurity.SimpleWebServer.dServerSocket
  - Normal confidence (1)

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

- Assignment 10 Understanding and Using Static Code
- Demo
- Oops
- Spotbugs (2)
  - JRE System Library [JavaSE-1.8]
  - src (2)
    - com.learnsecurity (2)
    - SimpleWebServer.java (2)

SimpleWebServer.java

```

59  String pathname = null;
60
61  /* parse the HTTP request */
62  StringTokenizer st =
63      new StringTokenizer (request, " ");
64
65  command = st.nextToken();
66  pathname = st.nextToken();
67
68  if (command.equals("GET")) {
69      /* if the request is a GET
       try to respond with the file
       the user is requesting */
70      serveFile (osw,pathname);
71
72  }

```

Bug Info

SimpleWebServer.java: 63

Navigation

Dereference of the result of readLine() without nullcheck in com.learnsecurity.SimpleWebServer.processRequest(Socket)

Value loaded from request

**Bug:** Dereference of the result of readLine() without nullcheck in com.learnsecurity.SimpleWebServer.processRequest(Socket)

The result of invoking readLine() is dereferenced without checking to see if the result is null. If there are no more lines of text to read, readLine() will return null and dereferencing that will generate a null pointer exception.

**Rank:** Of Concern (15), **confidence:** Normal  
**Pattern:** NP\_DEREFERENCE\_OF\_READLINE\_VALUE  
**Type:** NP, **Category:** STYLE (Dodgy code)

Problems Javadoc Declaration Bug Explorer

Spotbugs (2)
 

- Of Concern (2)
  - High confidence (1)
    - Write to static field from instance method (1)
      - Write to static field com.learnsecurity.SimpleWebServer.dServerSocket
  - Normal confidence (1)
    - Dereference of the result of readLine() without nullcheck (1)
      - Dereference of the result of readLine() without nullcheck in com.learnsecurity.SimpleWebServer.processRequest(Socket)

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java

```
26  public SimpleWebServer() throws Exception {
27      dServerSocket = new ServerSocket(PORT);
28  }
29
30  public void run() throws Exception {
31      while (true) {
32          /* wait for a connection from a client */
33          Socket s = dServerSocket.accept();
34
35          /* then process the client's request */
36          processRequest(s);
37      }
38  }
39
40  /*
41   * Reads the HTTP request from the client, and
42   * responds with the file the user requested or
43   * a HTTP error code.
44  */
```

Bug Explorer

Spotbugs (2)

- Of Concern (2)
  - High confidence (1)
    - Write to static field from instance method (1)
      - Write to static field com.learnsecurity.SimpleWebServer.dServerSocket from instance method new com.learnsecurity.SimpleWebServer() [Of Concern(15), High confidence]
    - Normal confidence (1)
      - Dereference of the result of readLine() without nullcheck (1)
        - Dereference of the result of readLine() without nullcheck in com.learnsecurity.SimpleWebServer.processRequest(Socket) [Of Concern(15), Normal confidence]

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java

```
19  public class SimpleWebServer {
20      /* Run the HTTP server on this TCP port. */
21      private static final int PORT = 8080;
22
23      /* The socket used to process incoming connections
24       * from web clients */
25      private static ServerSocket dServerSocket;
26
27      public SimpleWebServer() throws Exception {
28          dServerSocket = new ServerSocket(PORT);
29      }
30
31      public void run() throws Exception {
32          while (true) {
33              /* wait for a connection from a client */
34              Socket s = dServerSocket.accept();
```

Bug Explorer

Spotbugs (2)

- Of Concern (2)
  - High confidence (1)
    - Write to static field from instance method (1)
      - Write to static field com.learnsecurity.SimpleWebServer.dServerSocket from instance method new com.learnsecurity.SimpleWebServer() [Of Concern(15), High confidence]
    - Normal confidence (1)
      - Dereference of the result of readLine() without nullcheck (1)
        - Dereference of the result of readLine() without nullcheck in com.learnsecurity.SimpleWebServer.processRequest(Socket) [Of Concern(15), Normal confidence]

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java

```

58     String command = null;
59     String pathname = null;
60
61     /* parse the HTTP request */
62     StringTokenizer st =
63         new StringTokenizer (request, " ");
64
65     command = st.nextToken();
66     pathname = st.nextToken();
67
68     if (command.equals("GET")) {
69         /* if the request is a GET
70          * try to respond with the file
71          * the user is requesting */
72         serveFile (osw,pathname);
73     }
74     else {

```

Bug Explorer

- Spotbugs (2)
  - Of Concern (2)
    - High confidence (1)
      - Write to static field from instance method (1)
        - Write to static field com.learnsecurity.SimpleWebServer.dServerSocket from instance method new com.learnsecurity.SimpleWebServer() [Of Concern(15), High confidence]
    - Normal confidence (1)
      - Dereference of the result of readLine() without nullcheck (1)
        - Dereference of the result of readLine() without nullcheck in com.learnsecurity.SimpleWebServer.processRequest(Socket) [Of Concern(15), Normal confidence]

XML Report:



### a) Part 2 - SonarLint

Security rules enabled (along with required security plugins):

Window => Preferences

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

New Window

Editor > name = null;

Appearance > HTTP request \*/

Show View > zer st =

Perspective > ngTokenizer (request, " ");

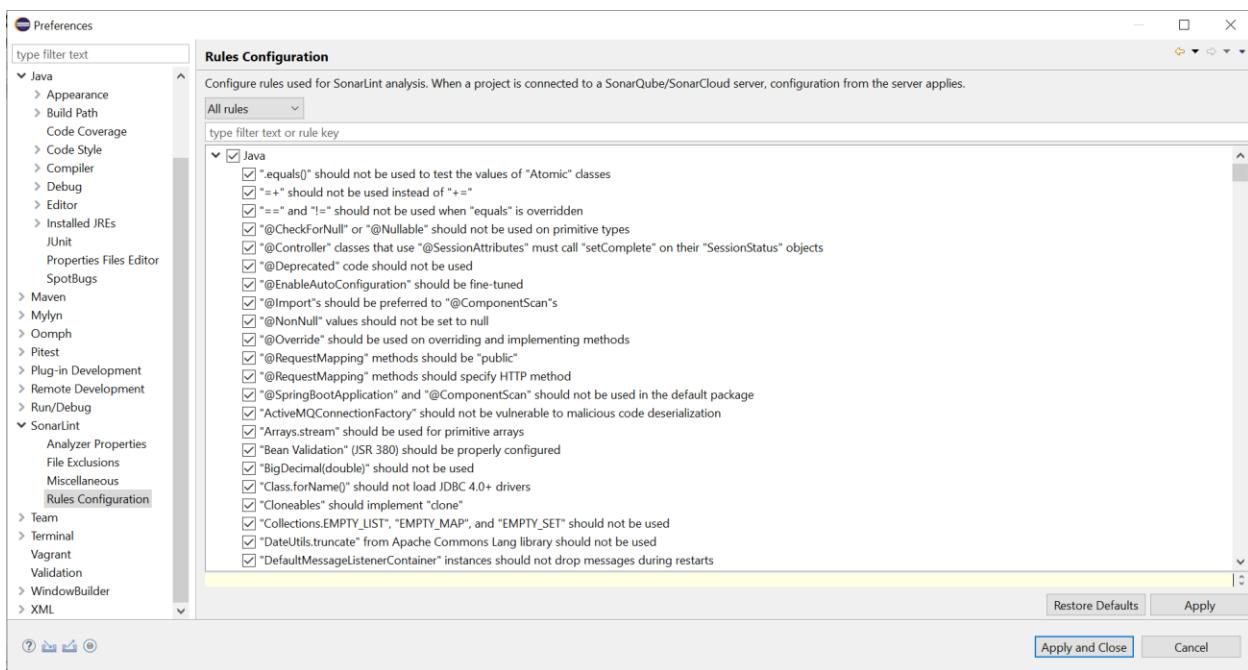
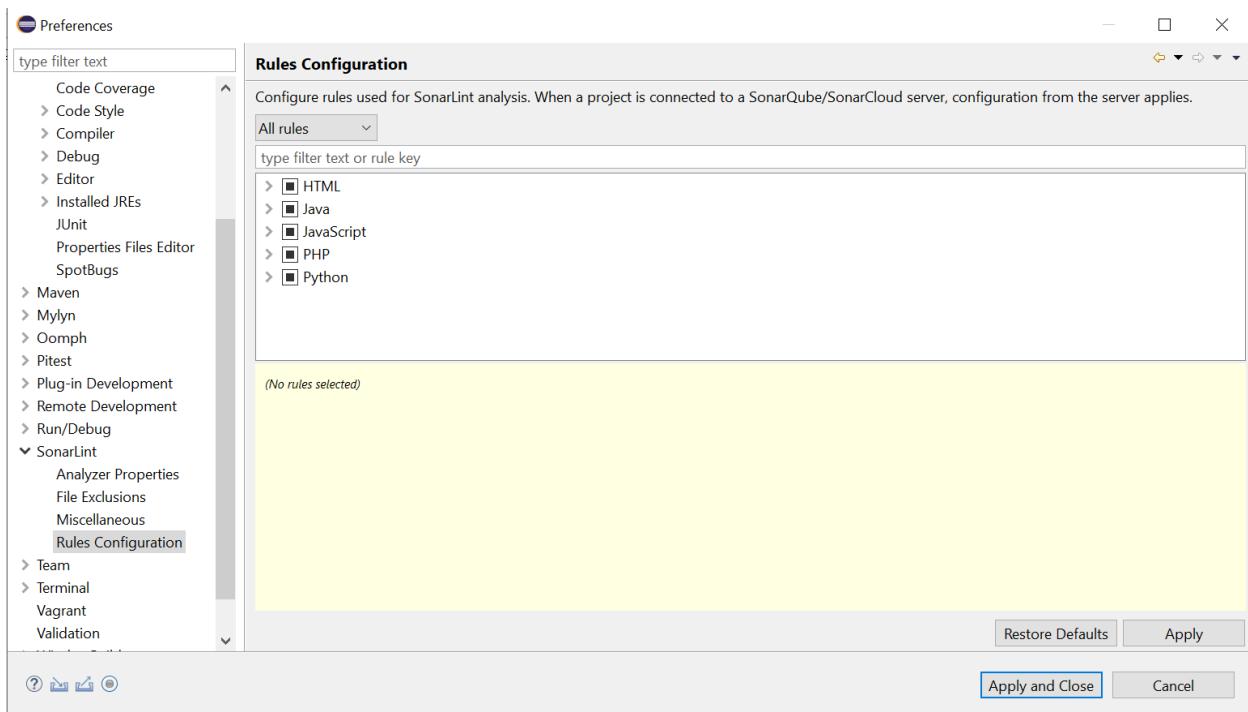
Navigation > .nextToken();

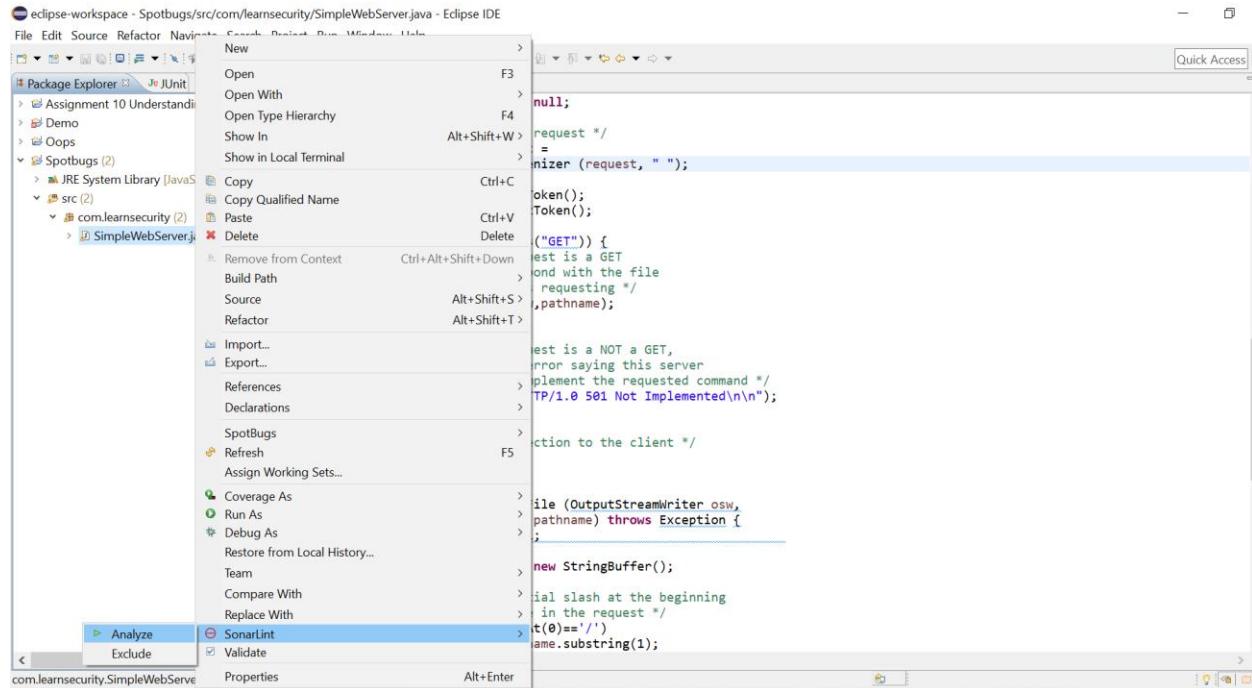
Preferences > pathname = st.nextToken();

```

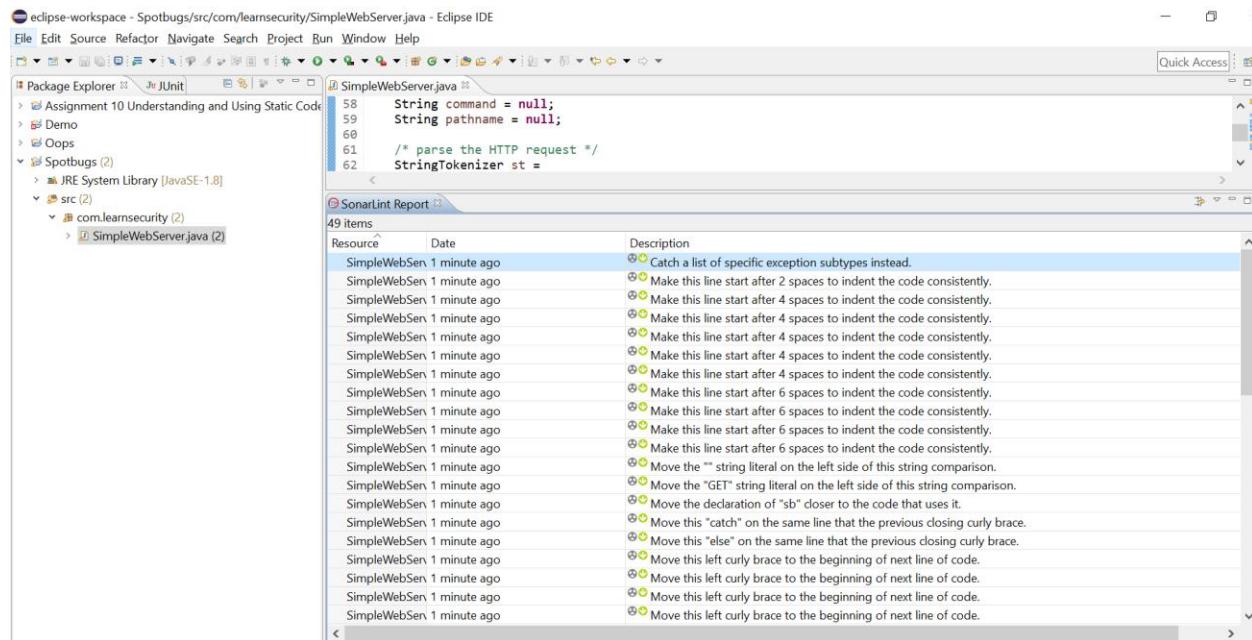
66     command = st.nextToken();
67     pathname = st.nextToken();
68     if (command.equals("GET")) {
69         /* if the request is a GET
70          * try to respond with the file
71          * the user is requesting */
72         serveFile (osw,pathname);
73     }
74     else {

```

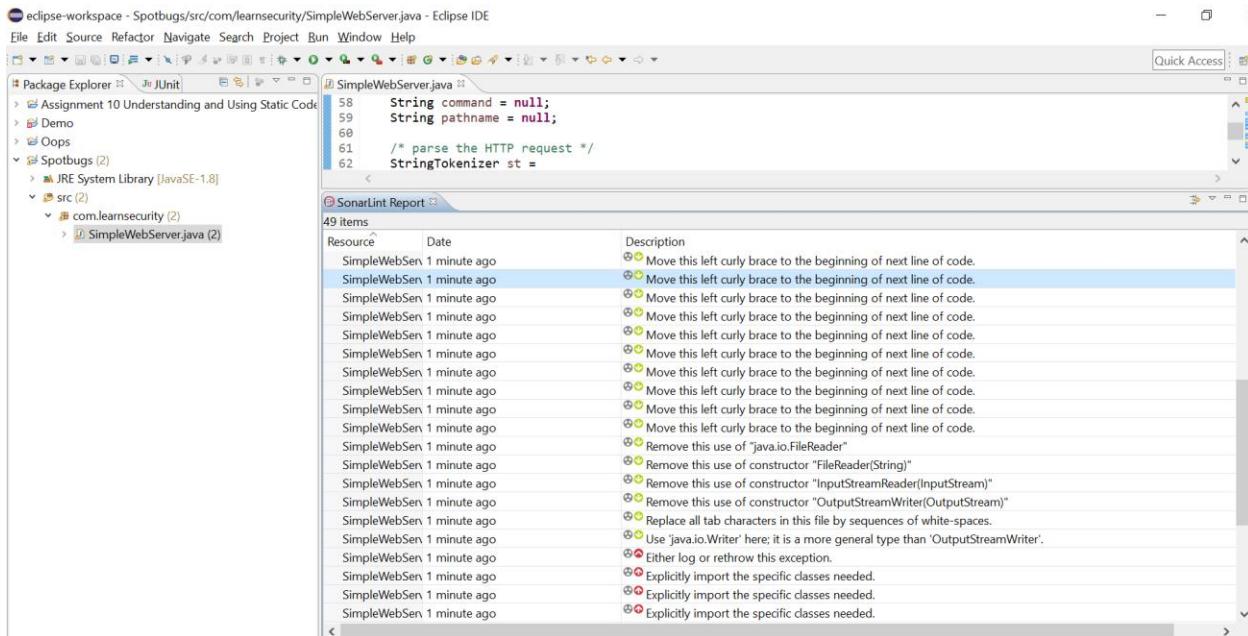
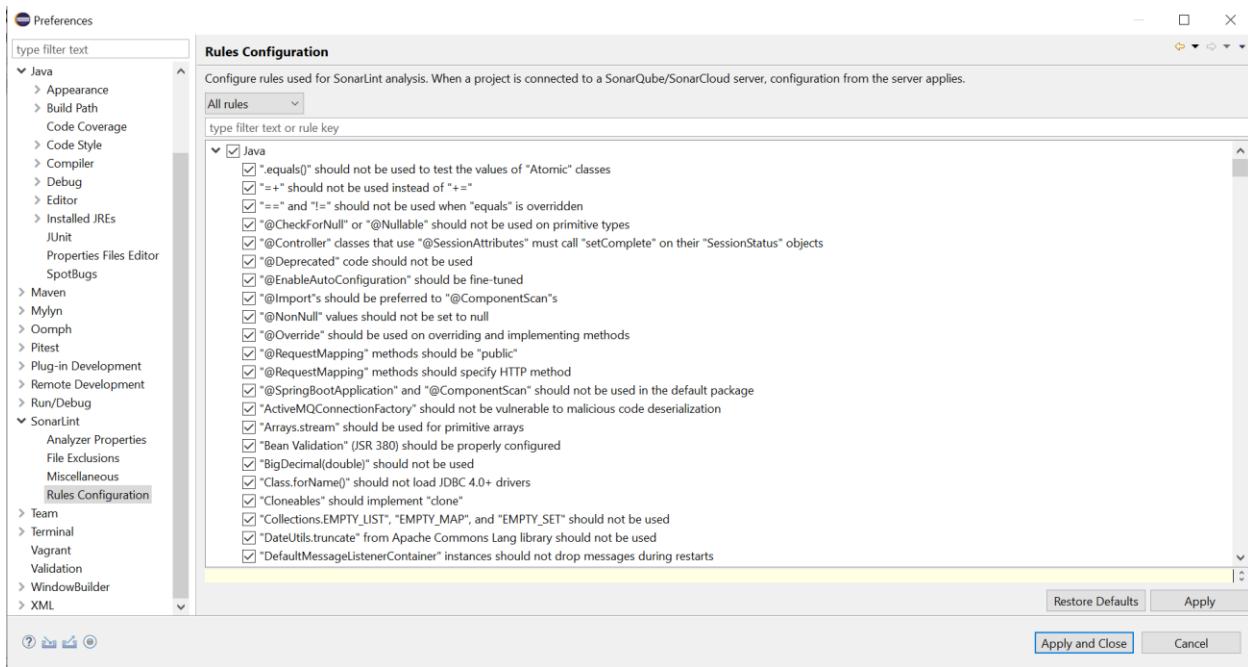




## Raw results provided (in separate file, directly output from tool):



## Turned on most aggressive mode in tool for finding defects:



eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java

```
58     String command = null;
59     String pathname = null;
60
61     /* parse the HTTP request */
62     StringTokenizer st =
```

SonarLint Report

49 items

Resource	Date	Description
SimpleWebSen	1 minute ago	Remove this use of constructor "FileReader(String)"
SimpleWebSen	1 minute ago	Remove this use of constructor "InputStreamReader(InputStream)"
SimpleWebSen	1 minute ago	Remove this use of constructor "OutputStreamWriter(OutputStream)"
SimpleWebSen	1 minute ago	Replace all tab characters in this file by sequences of white-spaces.
SimpleWebSen	1 minute ago	Use 'java.io.Writer' here; it is a more general type than 'OutputStreamWriter'.
SimpleWebSen	1 minute ago	Either log or rethrow this exception.
SimpleWebSen	1 minute ago	Explicitly import the specific classes needed.
SimpleWebSen	1 minute ago	Explicitly import the specific classes needed.
SimpleWebSen	1 minute ago	Explicitly import the specific classes needed.
SimpleWebSen	1 minute ago	Missing curly brace.
SimpleWebSen	1 minute ago	Missing curly brace.
SimpleWebSen	1 minute ago	Add or update the header of this file.
SimpleWebSen	1 minute ago	Remove this throws clause.
SimpleWebSen		Move the array designator from the variable to the type.
SimpleWebSen		Define and throw a dedicated exception instead of using a generic one.
SimpleWebSen		Define and throw a dedicated exception instead of using a generic one.
SimpleWebSen		Remove this assignment of "dServerSocket".
SimpleWebSen		Replace the synchronized class "StringBuffer" by an unsynchronized one such as "StringBuilder".
SimpleWebSen		Add an end condition to this loop.
SimpleWebSen		Use try-with-resources or close this "FileReader" in a "finally" clause.

File /Spotbugs/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 19:51)

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java

```
30 }
31
32 public void run() throws Exception {
33     while (true) {
34         /* wait for a connection from a client */
35         Socket s = dServerSocket.accept();
36
37         /* then process the client's request */
38         processRequest(s);
39     }
40 }
```

SonarLint Report

49 items

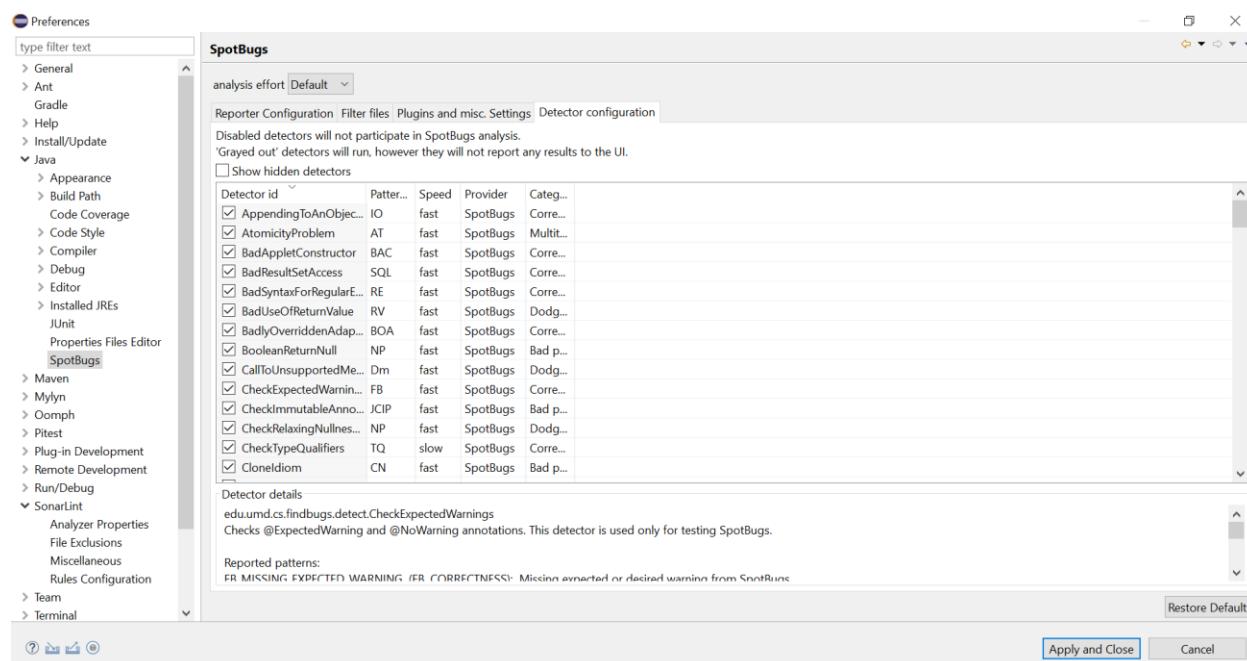
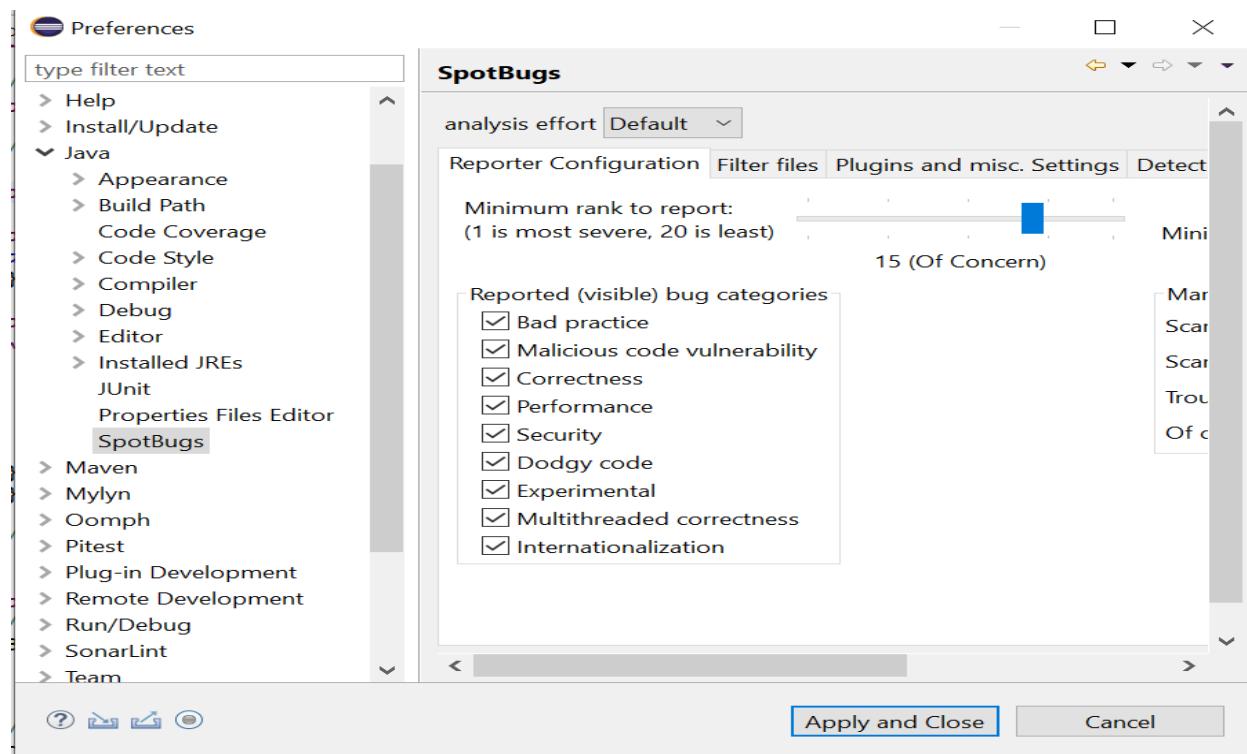
Resource	Date	Description
SimpleWebSen	1 minute ago	Either log or rethrow this exception.
SimpleWebSen	1 minute ago	Explicitly import the specific classes needed.
SimpleWebSen	1 minute ago	Explicitly import the specific classes needed.
SimpleWebSen	1 minute ago	Explicitly import the specific classes needed.
SimpleWebSen	1 minute ago	Missing curly brace.
SimpleWebSen	1 minute ago	Missing curly brace.
SimpleWebSen	1 minute ago	Add or update the header of this file.
SimpleWebSen	1 minute ago	Remove this throws clause.
SimpleWebSen		Move the array designator from the variable to the type.
SimpleWebSen		Define and throw a dedicated exception instead of using a generic one.
SimpleWebSen		Define and throw a dedicated exception instead of using a generic one.
SimpleWebSen		Remove this assignment of "dServerSocket".
SimpleWebSen		Replace the synchronized class "StringBuffer" by an unsynchronized one such as "StringBuilder".
SimpleWebSen		Add an end condition to this loop.
SimpleWebSen		Use try-with-resources or close this "FileReader" in a "finally" clause.

File /Spotbugs/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 19:51)

## a) Part 2 – SpotBugs

Security rules enabled (along with required security plugins):

Navigate to Window => Preferences



eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

The screenshot shows the Eclipse IDE interface with the file `Register.java` open in the editor. A context menu is displayed, and the option `SpotBugs` is highlighted.

```

    package com.learnsecurity;
    import java.util.ArrayList;
    public class Register {
        /**
         * Check input value. */
        if (this.typeOfUser == "Individual") {
            System.out.println("This is valid individual user");
        } else if (this.typeOfUser == "Business") {
            System.out.println("This is valid Business user");
        }
    }
  
```

### Raw results provided (in separate file, directly output from tool):

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

A modal dialog box titled "SpotBugs analysis finished" is displayed, stating "2 warnings found". It includes a checkbox "Remember the choice and do not ask me in the future" and three buttons: "Yes", "No", and "Cancel".

```

    package com.learnsecurity;
    import java.util.ArrayList;
    public class Register {
        /**
         * Check input value. */
        if (this.typeOfUser == "Individual") {
            System.out.println("This is valid individual user");
        } else if (this.typeOfUser == "Business") {
            System.out.println("This is valid Business user");
        }
    }
  
```

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit SimpleWebServer.java Register.java

```
1 package com.learnsecurity;
2
3 import java.util.ArrayList;
4 /**
5  * Class to Register an individual
6  */
7 public class Register {
8     /** The check input value. */
9     // This should be "Individual" or "Business" or "Admin"
10    private String typeOfUser= null;
11    // Name of the user
12    private String name = null;
13
14    /**
15     * The Constructor.
16     */
17    public Register() {
18        this.typeOfUser = null;
19        this.name = null;
20    }
21    public void run() {
22        individualInput();
23        IndividualValidation();
24        ArrayList<String> list = new ArrayList<>();
25        list.add(this.typeOfUser);
26        list.add(this.name);
27        System.out.println("Information of the name and type of user: "+list.get(2));
28    }
29    /**
30     * Check the input given in the individual field is for Individual user.
31     */
32    private void individualInput() {
33        if (this.typeOfUser == "Individual") {
34            System.out.println("This is valid individual user");
35        } else if (this.typeOfUser == "Business") {
36            System.out.println("This is valid Business user");
37        }
38    }
39
40    /**
41     * Validate the Individual and give information where to find the details.
42     */
43    private static void IndividualValidation() {
44        boolean dropDown = false;
45        if (dropDown == true) {
46            System.out.println("You can check the complete details in the Information page");
47        } else {
48            System.exit(0);
49        }
50    }
51 }
```

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit SimpleWebServer.java Register.java

```
17    public Register() {
18        this.typeOfUser = null;
19        this.name = null;
20    }
21    public void run() {
22        individualInput();
23        IndividualValidation();
24        ArrayList<String> list = new ArrayList<>();
25        list.add(this.typeOfUser);
26        list.add(this.name);
27        System.out.println("Information of the name and type of user: "+list.get(2));
28    }
29    /**
30     * Check the input given in the individual field is for Individual user.
31     */
32    private void individualInput() {
33        if (this.typeOfUser == "Individual") {
34            System.out.println("This is valid individual user");
35        } else if (this.typeOfUser == "Business") {
36            System.out.println("This is valid Business user");
37        } else return;
38    }
39
40    /**
41     * Validate the Individual and give information where to find the details.
42     */
43    private static void IndividualValidation() {
44        boolean dropDown = false;
45        if (dropDown == true) {
46            System.out.println("You can check the complete details in the Information page");
47        } else {
48            System.exit(0);
49        }
50    }
51 }
```

## Turned on most aggressive mode in tool for finding defects:

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java Register.java

```

37     } else return;
38   }
39
40  /**
41   * Validate the Individual and give information where to find the details.
42   */
43  private static void IndividualValidation() {
44    boolean dropdown = false;
45    if (dropdown = true) {
46      System.out.println("You can check the complete details in the Information page");
47    } else {
48      System.exit(0);
49    }
50  }
51 }
```

Bug Explorer

- Spotbugs (5)
  - Scary (1)
    - High confidence (1)
      - Method assigns boolean literal in boolean expression (1)
        - com.learnsecurity.Register.IndividualValidation() assigns boolean literal in boolean expression [Scary(5), High confidence]
  - Troubling (2)
    - Normal confidence (2)
      - Field only ever set to null (2)
        - Field only ever set to null: com.learnsecurity.Register.name [Troubling(12), Normal confidence]
        - Field only ever set to null: com.learnsecurity.Register.typeOfUser [Troubling(12), Normal confidence]
  - Of Concern (2)
    - High confidence (1)
      - Write to static field from instance method (1)
    - Normal confidence (1)
      - Dereference of the result of reading() without nullcheck (1)

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java Register.java

```

4 /**
5  * Class to Register an individual
6 */
7 public class Register {
8   /** The check input value. */
9   // This should be "Individual" or "Business" or "Admin"
10  private String typeOfUser= null;
11  // Name of the user
12  private String name = null;
13
14  /**
15   * The Constructor.
16   */
17  public Register() {
18    this.typeOfUser = null;
19    this.name = null;
20 }
```

Bug Explorer

- Spotbugs (5)
  - Scary (1)
    - High confidence (1)
      - Method assigns boolean literal in boolean expression (1)
        - com.learnsecurity.Register.IndividualValidation() assigns boolean literal in boolean expression [Scary(5), High confidence]
  - Troubling (2)
    - Normal confidence (2)
      - Field only ever set to null (2)
        - Field only ever set to null: com.learnsecurity.Register.name [Troubling(12), Normal confidence]
        - Field only ever set to null: com.learnsecurity.Register.typeOfUser [Troubling(12), Normal confidence]
  - Of Concern (2)
    - High confidence (1)
      - Write to static field from instance method (1)
    - Normal confidence (1)
      - Dereference of the result of reading() without nullcheck (1)

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java Register.java

```

4 /**
5  * Class to Register an individual
6 */
7 public class Register {
8     /** The check input value. */
9     // This should be "Individual" or "Business" or "Admin"
10    private String typeOfUser= null;
11    // Name of the user
12    private String name = null;
13
14    /**
15     * The Constructor.
16     */
17    public Register() {
18        this.typeOfUser = null;
19        this.name = null;
20    }

```

Bug Explorer

- Spotbugs (5)
  - Scary (1)
    - High confidence (1)
      - Method assigns boolean literal in boolean expression (1)
        - com.learnsecurity.Register.IndividualValidation() assigns boolean literal in boolean expression [Scary(5), High confidence]
  - Troubling (2)
    - Normal confidence (2)
      - Field only ever set to null (2)
        - Field only ever set to null: com.learnsecurity.Register.name [Troubling(12), Normal confidence]
        - Field only ever set to null: com.learnsecurity.Register.typeOfUser [Troubling(12), Normal confidence]
  - Of Concern (2)
    - High confidence (1)
      - Write to static field from instance method (1)
    - Normal confidence (1)
      - Preference of the result of readInfl without nullcheck (1)

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Quick Access

Package Explorer JUnit

SimpleWebServer.java Register.java

```

40 /**
41  * Validate the Individual and give information where to find the details.
42 */
43 private static void IndividualValidation() {
44     boolean dropdown = false;
45     if (dropdown == true) {
46         System.out.println("You can check the complete details in the Information page");

```

Bug Info

Register.java: 45

Navigation

com.learnsecurity.Register.IndividualValidation() assigns boolean literal in boolean expression

**Bug:** com.learnsecurity.Register.IndividualValidation() assigns boolean literal in boolean expression

This method assigns a literal boolean value (true or false) to a boolean variable inside an if or while expression. Most probably this was supposed to be a boolean comparison using ==, not an assignment using =.

**Rank:** Scary (5), **confidence:** High  
**Pattern:** QBA\_QUESTIONABLE\_BOOLEAN\_ASSIGNMENT  
**Type:** QBA, **Category:** CORRECTNESS (Correctness)

---

**XML output:**

```

<BugInstance type="QBA_QUESTIONABLE_BOOLEAN_ASSIGNMENT" priority="1" rank="5" abbrev="QBA" category="CORRECTNESS">
    <Class classname="com.learnsecurity.Register">
        <SourceLine classname="com.learnsecurity.Register" sourcefile="Register.java" sourcepath="com/learnsecurity/Register.java" start="45" end="46" startBytecode="0" endBytecode="1" />
    </Class>
    <Method classname="com.learnsecurity.Register" name="IndividualValidation" signature="()V" isStatic="true">
        <SourceLine classname="com.learnsecurity.Register" start="44" end="50" startBytecode="0" endBytecode="98" sourcefile="Register.java" sourcepath="com/learnsecurity/Register.java" />
    </Method>
</BugInstance>

```

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java Register.java

```
8  /** The check input value. */
9  // This should be "Individual" or "Business" or "Admin"
10 private String typeOfUser= null;
11 // Name of the user
12 private String name = null;
13 /**
14 */

Bug Info
Register.java: 12
Navigation
Field only ever set to null: com.learnsecurity.Register.name
Field com.learnsecurity.Register.name

Bug: Field only ever set to null: com.learnsecurity.Register.name

All writes to this field are of the constant value null, and thus all reads of the field will return null. Check for errors, or remove it if it is useless.

Rank: Troubling (12), confidence: Normal
Pattern: UWF_NULL_FIELD
Type: UwF, Category: CORRECTNESS (Correctness)

XML output:
```

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java Register.java

```
8  /** The check input value. */
9  // This should be "Individual" or "Business" or "Admin"
10 private String typeOfUser= null;
11 // Name of the user
12 private String name = null;
13 /**
14 */

Bug Info
Register.java: 10
Navigation
Field only ever set to null: com.learnsecurity.Register.typeOfUser
Field com.learnsecurity.Register.typeOfUser

Bug: Field only ever set to null: com.learnsecurity.Register.typeOfUser

All writes to this field are of the constant value null, and thus all reads of the field will return null. Check for errors, or remove it if it is useless.

Rank: Troubling (12), confidence: Normal
Pattern: UWF_NULL_FIELD
Type: UwF, Category: CORRECTNESS (Correctness)

XML output:
```

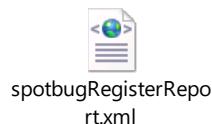
eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

```

File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer JUnit Quick Access
SimpleWebServer.java Register.java
1 package com.learnsecurity;
2 import java.util.ArrayList;
3 /**
4  * Class to Register an individual
5  */
6 public class Register {
7     /** The check input value. */
8     // This should be "Individual" or "Business" or "Admin"
9     private String typeOfUser= null;
10    // Name of the user
11    private String name = null;
12
13    /**
14     * The Constructor.
15     */
16    public Register() {
17        this.typeOfUser = null;
18        this.name = null;
19    }
20    public void run() {
21        individualInput();
22        IndividualValidation();
23        ArrayList<String> list = new ArrayList<>();
24        list.add(this.typeOfUser);
25        list.add(this.name);
26        System.out.println("Information of the name and type of user: "+list.get(2));
27    }
28    /**
29     * Check the input given in the individual field is for Individual user.
30     */
31    private void individualInput() {
32        if (this.typeOfUser == "Individual") {
33            System.out.println("This is valid individual user");
34        } else if (this.typeOfUser == "Business") {
35            System.out.println("This is valid Business user");
36        }
}

```

XML Report:



## b) Part 2 – SonarLint

**Security rules enabled (along with required security plugins):**

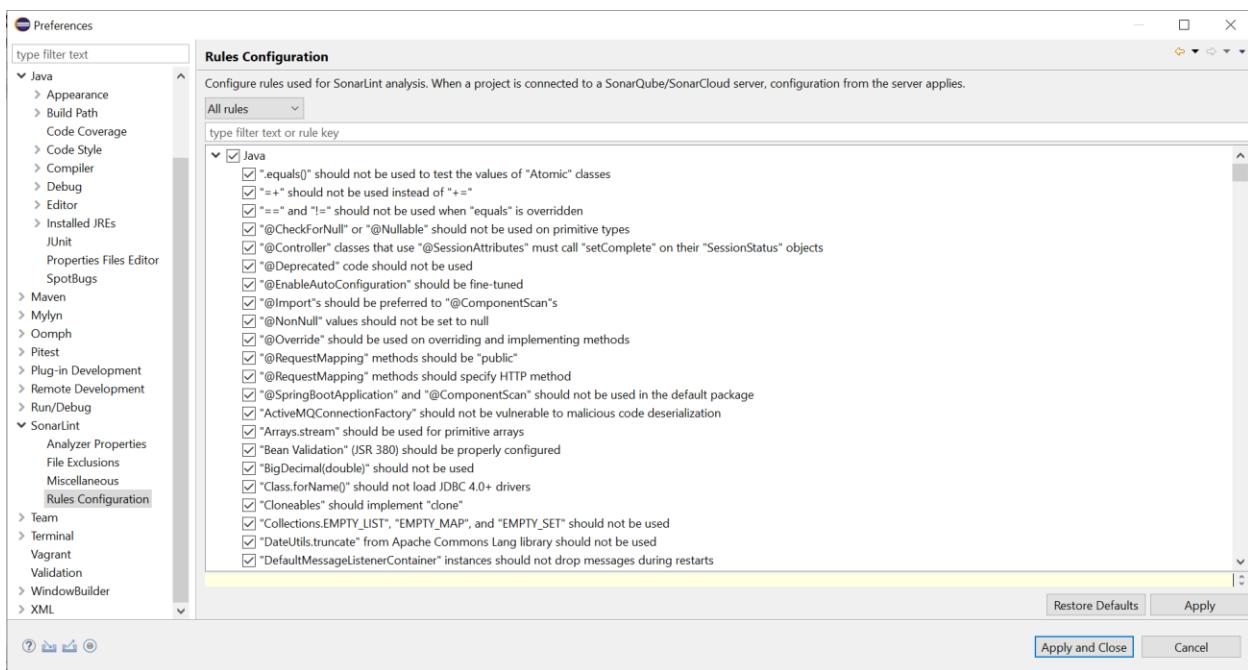
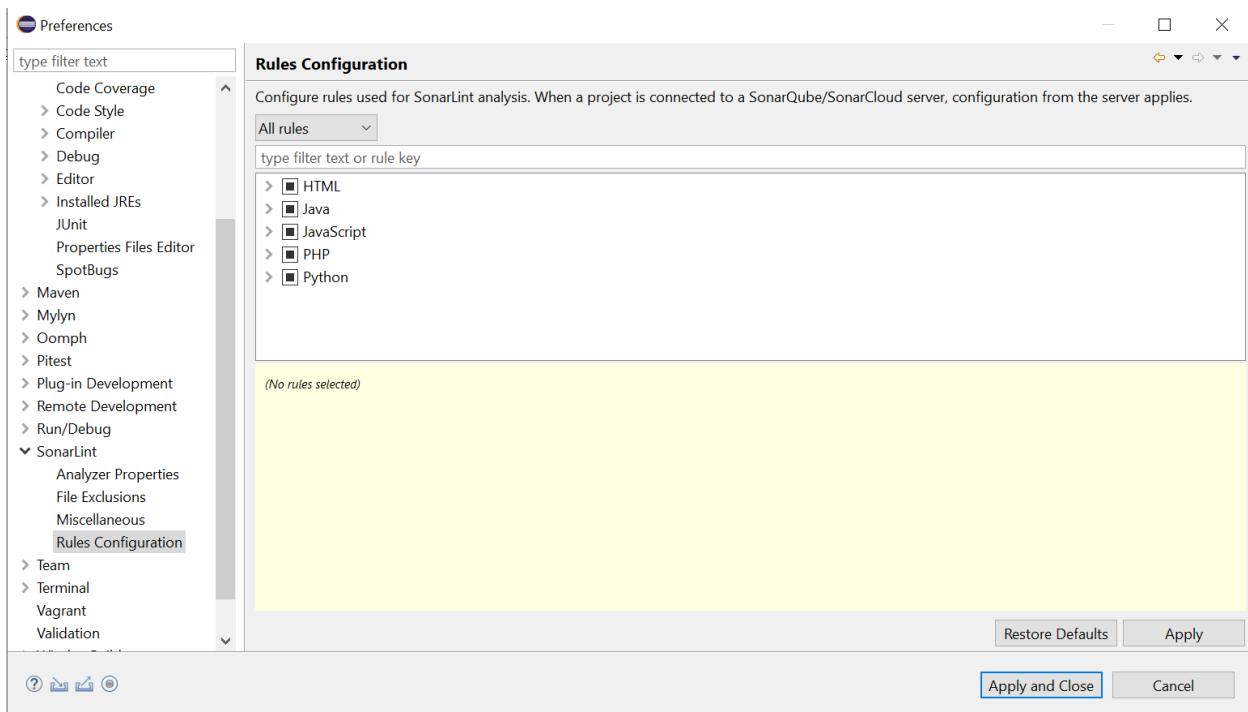
Navigate to Window => Preferences

eclipse-workspace - Spotbugs/src/com/learnsecurity/SimpleWebServer.java - Eclipse IDE

```

File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer JUnit
New Window
Editor
Appearance
Show View
Perspective
Navigation
Preferences
66    pathname = st.nextToken();
67
68    if (command.equals("GET")) {
69        /* if the request is a GET

```



eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- Assignment 10
- Demo
- Oops
- Spotbugs (5)
  - JRE System Library
  - src (5)
    - com.learnsecurity
    - Register
    - SimpleWebServer

New

Open F3

Open With F4

Show In Alt+Shift+W

Ctrl+C

Copy Ctrl+V

Paste Ctrl+X

Delete Ctrl+Shift+D

Remove from Context Ctrl+Alt+Shift+Down

Build Path

Source Alt+Shift+S

Refactor Alt+Shift+T

Import... Ctrl+Shift+I

Export... Ctrl+Shift+E

References

Declarations

SpotBugs

Refresh F5

Assign Working Sets...

Coverage As

Run As

Debug As

Restore from Local History...

Team

Compare With

Replace With

SonarLint

Validate

Properties

Alt + Enter

```

 1 package com.learnsecurity;
 2
 3 import java.util.ArrayList;
 4
 5 public class Register {
 6     /**
 7      * check input value. */
 8     public void register() {
 9         if (this.typeOfUser == null) {
10             System.out.println("The user name is null");
11         }
12         if (this.name == null) {
13             System.out.println("The name is null");
14         }
15         if (this.typeOfUser == null) {
16             System.out.println("The type of user is null");
17         }
18         if (this.name == null) {
19             System.out.println("The name is null");
20         }
21         if (this.typeOfUser == null) {
22             System.out.println("The type of user is null");
23         }
24         if (this.name == null) {
25             System.out.println("The name is null");
26         }
27         if (this.typeOfUser == null) {
28             System.out.println("The type of user is null");
29         }
30         if (this.name == null) {
31             System.out.println("The name is null");
32         }
33         if (this.typeOfUser == null) {
34             System.out.println("The type of user is null");
35         }
36         if (this.name == null) {
37             System.out.println("The name is null");
38         }
39         if (this.typeOfUser == null) {
40             System.out.println("The type of user is null");
41         }
42         if (this.name == null) {
43             System.out.println("The name is null");
44         }
45         if (this.typeOfUser == null) {
46             System.out.println("The type of user is null");
47         }
48         if (this.name == null) {
49             System.out.println("The name is null");
50         }
51         if (this.typeOfUser == null) {
52             System.out.println("The type of user is null");
53         }
54         if (this.name == null) {
55             System.out.println("The name is null");
56         }
57         if (this.typeOfUser == null) {
58             System.out.println("The type of user is null");
59         }
60         if (this.name == null) {
61             System.out.println("The name is null");
62         }
63         if (this.typeOfUser == null) {
64             System.out.println("The type of user is null");
65         }
66         if (this.name == null) {
67             System.out.println("The name is null");
68         }
69         if (this.typeOfUser == null) {
70             System.out.println("The type of user is null");
71         }
72         if (this.name == null) {
73             System.out.println("The name is null");
74         }
75         if (this.typeOfUser == null) {
76             System.out.println("The type of user is null");
77         }
78         if (this.name == null) {
79             System.out.println("The name is null");
80         }
81         if (this.typeOfUser == null) {
82             System.out.println("The type of user is null");
83         }
84         if (this.name == null) {
85             System.out.println("The name is null");
86         }
87         if (this.typeOfUser == null) {
88             System.out.println("The type of user is null");
89         }
90         if (this.name == null) {
91             System.out.println("The name is null");
92         }
93         if (this.typeOfUser == null) {
94             System.out.println("The type of user is null");
95         }
96         if (this.name == null) {
97             System.out.println("The name is null");
98         }
99         if (this.typeOfUser == null) {
100            System.out.println("The type of user is null");
101        }
102    }
103 }

```

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

- Assignment 10 Understanding and Using Static Code
- Demo
- Oops
- Spotbugs (5)
  - JRE System Library [JavaSE-1.8]
  - src (5)
    - com.learnsecurity (3)
      - Register.java
      - SimpleWebServer.java

SonarLint Report

39 items

Resource	Date	Description
Register.java	4 hours ago	Make this line start after 2 spaces to indent the code consistently.
Register.java	19 hours ago	Remove this initialization to "null", the compiler will do that for you.
Register.java	19 hours ago	Assign this magic number 2 to a well-named constant, and use the constant instead.
Register.java	19 hours ago	Replace this use of System.out or System.err by a logger.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	19 hours ago	Move this "else" keyword to a new dedicated line.
Register.java	19 hours ago	Remove this redundant jump.
Register.java	19 hours ago	Missing curly brace.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Replace all tab characters in this file by sequences of white-spaces.
Register.java	19 hours ago	Remove this useless assignment to local variable "dropDown".
Register.java	19 hours ago	Change this condition so that it does not always evaluate to "true"
Register.java	20 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Move this "else" keyword to a new dedicated line.

File /Spotbugs/src/com/learnsecurity/Register.java (at 26/04/2020 16:13)

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

Assignment 10 Understanding and Using Static Code

Demo

Oops

Spotbugs (5)

JRE System Library [JavaSE-1.8]

src (5)

- com.learnsecurity (5)
  - Register.java (3)
  - SimpleWebServer.java (2)

SimpleWebServer.java Register.java

```
7 public class Register {
8     /** The check input value. */
9 }
```

SonarLint Report

39 items

Resource	Date	Description
Register.java	20 hours ago	Move this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Move this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Move this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Move this "else" keyword to a new dedicated line.
Register.java	20 hours ago	Move this "else" keyword to a new dedicated line.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Remove this initialization to "null", the compiler will do that for you.
Register.java	20 hours ago	Rename this method name to match the regular expression `^a-zA-Z0-9*\$`.
Register.java	20 hours ago	Extract the assignment out of this expression.
Register.java	20 hours ago	Replace this use of System.out or System.err by a logger.
Register.java	20 hours ago	Replace this use of System.out or System.err by a logger.
Register.java	20 hours ago	Replace this use of System.out or System.err by a logger.
Register.java	20 hours ago	Remove this call to "exit" or ensure it is really required.
Register.java	20 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	20 hours ago	Add or update the header of this file.

File /Spotbugs/src/com/learnsecurity/Register.java (at 26/04/2020 16:13)

Use the "equals" method if value comparison was intended.

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

Assignment 10 Understanding and Using Static Code

Demo

Oops

Spotbugs (5)

JRE System Library [JavaSE-1.8]

src (5)

- com.learnsecurity (5)
  - Register.java (3)
  - SimpleWebServer.java (2)

SimpleWebServer.java Register.java

```
28 }
29 /**
30  * Check the input given in the individual field is for Individual user.
31 */
32 private void individualInput() {
33     if (this.typeOfUser == "Individual") {
34         System.out.println("This is valid individual user");
35     } else if (this.typeOfUser == "Business") {
36         System.out.println("This is valid Business user");
37     } else return;
38 }
39 /**
40 */

SonarLint Report
```

39 items

Resource	Date	Description
Register.java	4 hours ago	Make this line start after 2 spaces to indent the code consistently.
Register.java	19 hours ago	Remove this initialization to "null", the compiler will do that for you.
Register.java	19 hours ago	Assign this magic number 2 to a well-named constant, and use the constant instead.
Register.java	19 hours ago	Replace this use of System.out or System.err by a logger.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	19 hours ago	Move this "else" keyword to a new dedicated line.
Register.java	19 hours ago	Remove this redundant jump.
Register.java	19 hours ago	Missing curly brace.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Replace all tab characters in this file by sequences of white-spaces.

File /Spotbugs/src/com/learnsecurity/Register.java (at 26/04/2020 16:13)

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java Register.java

```

28 }
29 /**
30  * Check the input given in the individual field is for Individual user.
31 */
32 private void individualInput() {
33     if (this.typeOfUser == "Individual") {
34         System.out.println("This is valid individual user");
35     } else if (this.typeOfUser == "Business") {
36         System.out.println("This is valid Business user");
37     } else return;
38 }
39 /**
40 */

```

SonarLint Report

39 items

Resource	Date	Description
Register.java	4 hours ago	Make this line start after 2 spaces to indent the code consistently.
Register.java	19 hours ago	Remove this initialization to "null", the compiler will do that for you.
Register.java	19 hours ago	Assign this magic number 2 to a well-named constant, and use the constant instead.
Register.java	19 hours ago	Replace this use of System.out or System.err by a logger.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	19 hours ago	Move this "else" keyword to a new dedicated line.
Register.java	19 hours ago	Remove this redundant jump.
Register.java	19 hours ago	Missing curly brace.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Replace all tab characters in this file by sequences of white-spaces.

File /Spotbugs/src/com/learnsecurity/Register.java (at 26/04/2020 16:13)

eclipse-workspace - Spotbugs/src/com/learnsecurity/Register.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit

SimpleWebServer.java Register.java

```

40 /**
41  * Validate the Individual and give information where to find the details.
42 */
43 private static void IndividualValidation() {
44     boolean dropdown = false;
45     if (dropdown == true) {
46         System.out.println("You can check the complete details in the Information page");
47     } else {
48         System.exit(0);
49     }
50 }
51

```

SonarLint Report

39 items

Resource	Date	Description
Register.java	19 hours ago	Move this left curly brace to the beginning of next line of code.
Register.java	19 hours ago	Move this "else" keyword to a new dedicated line.
Register.java	19 hours ago	Remove this redundant jump.
Register.java	19 hours ago	Missing curly brace.
Register.java	19 hours ago	Use the "equals" method if value comparison was intended.
Register.java	19 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	19 hours ago	Replace all tab characters in this file by sequences of white-spaces.
Register.java	19 hours ago	Remove this useless assignment to local variable "dropDown".
Register.java	19 hours ago	Change this condition so that it does not always evaluate to "true"
Register.java	20 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 4 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 6 spaces to indent the code consistently.
Register.java	20 hours ago	Make this line start after 6 spaces to indent the code consistently.

File /Spotbugs/src/com/learnsecurity/Register.java (at 26/04/2020 16:13)

## Reports:

	SpotbugsSimpleWeb Server report.xlsx		SonarLintSimpleWeb Server report.xlsx		SpotbugsRegister report.xlsx		SonarLintRegister report.xlsx
---	--------------------------------------	---	---------------------------------------	---	------------------------------	---	-------------------------------